1991

The use of conformal subdomain basis functions in the method of moments computations for a thin wire.

Walter, Bruce A.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

THE USE OF CONFORMAL SUBDOMAIN
BASIS FUNCTIONS IN THE METHOD OF MOMENTS
COMPUTATIONS FOR A THIN WIRE

by

Bruce A. Walter

December, 1991

Thesis Advisor: David C. Jenn

Approved for public release; distribution is unlimited.

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No 0704-0188

1a REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b RESTRICTIVE MARKINGS: NONE
2a SECURITY CLASSIFICATION AUTHORITY: N/A	3 DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution is unlimited.
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION: Naval Postgraduate School — 6b OFFICE SYMBOL (If applicable): 3A	7a. NAME OF MONITORING ORGANIZATION: Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code): Monterey, CA 93943-5000	7b ADDRESS (City, State, and ZIP Code): Monterey, CA 93943-5000
8a NAME OF FUNDING/SPONSORING ORGANIZATION — 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO — PROJECT NO — TASK NO — WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)

THE USE OF CONFORMAL SUBDOMAIN BASIS FUNCTIONS IN THE METHOD OF MOMENTS COMPUTATIONS FOR A THIN WIRE

12 PERSONAL AUTHOR(S)

Walter, Bruce A.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM ____ TO ____	1991, December	90

16 SUPPLEMENTARY NOTATION

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U. S. Government.

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Method of Moments
			Conformal Subdomain Basis Functions
			Thin Wire Integral Equation

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The purpose of this thesis is to investigate the use of Conformal Subdomain Basis Functions (CSBF) in the Method of Moments (MM) solution of a thin wire scatterer. The effect of using CSBF on the computed current and the scattered field is investigated by formulating and coding the MM solution for a thin wire loop and comparing the computed results for various loop sizes to measured data and two other MM codes. Significant reduction in the number of segments (and computer memory requirements) are found for loops with circumferences of less than one to two wavelengths for plane wave incidence. From these results, it is concluded that the use of CSBF will significantly reduce the number of segments required for the MM solution of a spiral antenna.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT: ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL: D. C. Jenn	22b TELEPHONE (Include Area Code): 408-646-2254 — 22c OFFICE SYMBOL: CODE EC/Jn

DD Form 1473, JUN 86 Previous editions are obsolete SECURITY CLASSIFICATION OF THIS PAGE

S/N 0102-LF-014-6603 Unclassified

THE USE OF CONFORMAL SUBDOMAIN BASIS FUNCTIONS IN THE METHOD OF MOMENTS COMPUTATIONS FOR A THIN WIRE

by

Bruce A. Walter
B.S.E.E., Virginia Polytechnic Institute, 1984

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December, 1991

Department of Electrical and Computer Engineering

ABSTRACT

The purpose of this thesis is to investigate the use of Conformal Subdomain Basis Functions (CSBF) in the Method of Moments (MM) solution of a thin wire scatterer. The effect of using CSBF on the computed current and the scattered field is investigated by formulating and coding the MM solution for a thin wire loop and comparing the computed results for various loop sizes to measured data and two other MM codes. Significant reduction in the number of segments (and computer memory requirements) are found for loops with circumferences of less than one to two wavelengths for plane wave incidence. From these results, it is concluded that the use of CSBF will significantly reduce the number of segments required for the MM solution of a spiral antenna.

TABLE OF CONTENTS

I. INTRODUCTION

Numerical techniques for solving electromagnetic scattering problems using integral equations and the method of moments (MM) are well known. The physical problem, specified by Maxwell's equations and boundary conditions, is reduced to an integro-differential equation over finite domains, and solved using a procedure referred to as the method of moments [Ref. 1]. The unknowns (usually currents) are represented by a series of basis functions with unknown expansion coefficients. The MM process generates a set of linear equations that must be solved simultaneously. Until recently, these techniques have been limited to small (1 to 10 wavelength) geometries because of computer run time and memory constraints. With the development of faster computers with more memory, the MM technique has increasing application to larger geometries. However, computer memory and run time can still be inadequate to solve many important antenna and scattering problems. Numerically efficient solutions require less computer memory and/or less computer run time. Therefore, any increase in the efficiency of a MM solution is of great practical interest.

The usual MM approach to modeling a thin wire of arbitrary shape is to specify a series of points, with piecewise linear segments between the points to approximate the wire. The current is represented by one or more basis functions, each with constant phase, over a piecewise linear segment. Typically, the size of the segments is set by how accurately the current or scattered field needs to be determined. For

convergence of the current, segment lengths of 0.05 λ to 0.1 λ are generally required.

A second factor that influences the segment size is the radius of curvature of the wire. Tightly curved wires require smaller segments to reproduce the wire shape accurately. When the radius of curvature is larger than a wavelength, the first case sets the segment size; when the radius of curvature is much less than a wavelength, the second case dictates the segment size (Figure 1).

All generally available MM codes based on the method of subdomains use the first approach. A natural question arises: Does dividing the wire into curved segments that conform to its shape, with arclengths restricted only by the maximum current variation rule, yield a solution that converges with fewer subsections? If the answer is yes, is the improvement in convergence worth the greater complexity and coding effort? To resolve this issue, the following approach is taken:

Figure 1. Left: Piecewise Linear Segments, Right: Curved Segments

1. Formulate the solution for linearly and circularly polarized plane wave incidence for a geometrically simple shape such as a loop.

2. Computer code the solution in FORTRAN.

3. Validate the solution using other MM solutions and measured data.

4. Study the convergence of the solution with respect to loop parameters and compare its performance to a method using linear subsections.

5. Study the effect of changes in program structure on its computational efficiency.

Chapter II discusses the derivation and the MM solution of the electric field integral equation for a thin wire loop. Chapter III discusses three MM FORTRAN programs used to determine the current on the loop. The entire domain solution (due to R. F. Harrington), which uses complex Fourier modes (HARLOOP) is considered to be the most accurate and therefore serves as a baseline for evaluating the other solutions. A second program that uses linear segments (LOOPSCAT) is compared to a third program that uses curved segments (CURVENEW). Chapter IV discusses the results obtained by the three methods and presents some guidelines in choosing an optimum solution method for a given antenna or scattering problem.

II. THE THIN WIRE INTEGRAL EQUATION

A. DERIVATION OF THE THIN-WIRE ELECTRIC FIELD INTEGRAL EQUATION

In this chapter, the integral equation for the current on a thin wire will be developed. Time-harmonic field quantities are assumed throughout. Phasor quantities are used with the $e^{j\omega t}$ dependency suppressed.

Referring to the thin wire geometry of Figure 2, the origin is point 0, the location of a source point is given by the vector **r'** and an observation point by the vector **r**. The wire radius, a, is considered constant over the length of the wire. The vector **l** is everywhere parallel to the surface of the wire. Since the sum of the tangential components of the incident and scattered electric field must vanish at the surface of a perfect electric conductor, the boundary condition is,

$$\hat{n} \times (E^i + E^s) = 0 \tag{2.1}$$

If the radius of the wire is small compared to the wavelength of the excitation, the surface current density, J_s, can be considered constant around the circumference of the wire and directed along its axis. The excitation field can be either an incident wave or an impressed voltage. The scattered field is the field due to the current on the conductor induced by the excitation field.

The wave equation in terms of the vector potential **A** is given by

$$\nabla^2 A + \beta^2 A = -\mu J_s \tag{2.2}$$

where $\beta = 2\pi/\lambda$. The solution to equation (2.2) is

$$A = \frac{\mu}{4\pi} \int_S \frac{J_s e^{j\beta|r-r'|}}{|r-r'|} dS' = \mu \int_S J_s \, g(r,r') dS' \tag{2.3}$$

where the integration is over the primed (source) coordinates. The Green's function, g(**r**,**r**') is defined as,

$$g(r,r') = \frac{e^{-j\beta|r-r'|}}{4\pi|r-r'|} . \tag{2.4}$$

The expression for the scattered electric field in terms of **A** is,

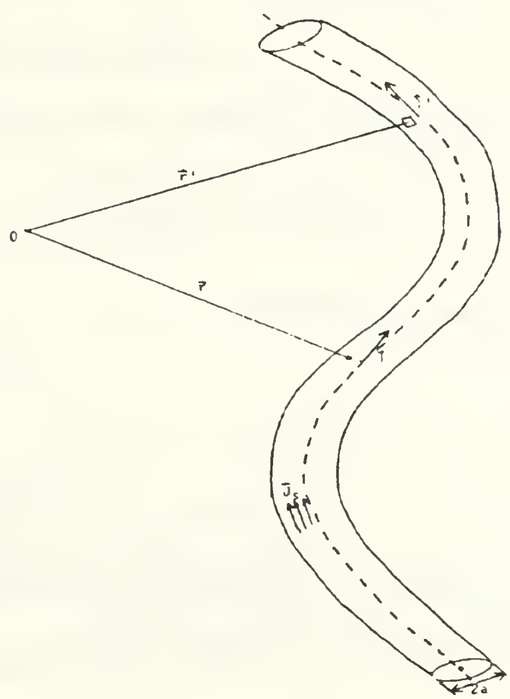


Figure 2. Thin Wire Geometry

$$\boldsymbol{E}^{s} = -j\omega \boldsymbol{A} - \frac{j}{\omega\mu\epsilon}\nabla(\nabla\cdot\boldsymbol{A}) \ . \tag{2.5}$$

Applying the boundary condition of equation (2.1),

$$\boldsymbol{E}^{i}_{\tan} = -\boldsymbol{E}^{s}_{\tan} = j\omega \boldsymbol{A} + \frac{j}{\omega\mu\epsilon}\nabla(\nabla\cdot\boldsymbol{A}) \qquad on\ S\ . \tag{2.6}$$

Substitution of **A** in equation (2.3) into equation (2.6) gives,

$$\boldsymbol{E}^{i}_{\tan} = j\omega\mu\int_{s'}\boldsymbol{J}_{s}g(\boldsymbol{r},\boldsymbol{r}')dS' + \frac{j}{\omega\mu\epsilon}\nabla\,\nabla\cdot\left[\mu\int_{s'}\boldsymbol{J}_{s}g(\boldsymbol{r},\boldsymbol{r}')dS'\right] \qquad on\ S. \tag{2.7}$$

Call the second term on the right side of equation (2.7) **V**, and assume the medium to be homogeneous,

$$\boldsymbol{V} = \nabla\left[\nabla\cdot\mu\int_{s'}\boldsymbol{J}_{s}\ g(\boldsymbol{r},\boldsymbol{r}')dS'\right] = \nabla\left[\mu\int_{s'}\nabla\cdot(\boldsymbol{J}_{s}\ g(\boldsymbol{r},\boldsymbol{r}'))dS'\right] . \tag{2.8}$$

The vector identity for the divergence of a scalar u times a vector **v** is,

$$\nabla\cdot(u\boldsymbol{v}) = \nabla u\cdot\boldsymbol{v} + u(\nabla\cdot\boldsymbol{v}) \ . \tag{2.9}$$

Applying this identity to equation (2.8) gives

$$\boldsymbol{V} = \nabla\left[\mu\int_{s'}(\nabla g(\boldsymbol{r},\boldsymbol{r}'))\cdot\boldsymbol{J}_{s}dS'\right] . \tag{2.10}$$

It can be shown that $\nabla g(\mathbf{r},\mathbf{r}') = -\nabla' g(\mathbf{r},\mathbf{r}')$ [Ref. 2]. Using this in equation (2.10) and applying the identity of equation (2.9) again yields,

$$\begin{aligned} \mathbf{V} &= -\nabla\left[\mu\int_{s'}(\nabla' g(\mathbf{r},\mathbf{r}'))\cdot \mathbf{J}_s dS'\right] \\ &= \nabla\left[\mu\int_{S'} g(\mathbf{r},\mathbf{r}')(\nabla'\cdot\mathbf{J}_s)dS' - \mu\int_{S'}\nabla'\cdot(g(\mathbf{r},\mathbf{r}')\mathbf{J}_s)dS'\right] \end{aligned} \tag{2.11}$$

where ∇' is the del operator defined with respect to the primed coordinates. The second integral on the right side of equation (2.11) is equal to zero by the surface divergence theorem [Ref. 3]. Thus, **V** simplifies to

$$\mathbf{V} = \mu\int_{S'}(\nabla'\cdot\mathbf{J}_s)\nabla g(\mathbf{r},\mathbf{r}')dS' . \tag{2.12}$$

Substitution of equation (2.12) into equation (2.7) gives an integral equation for $\mathbf{J}_s$,

$$\mathbf{E}^i_{\tan} = j\omega\mu\int_{S'}\mathbf{J}_s g(\mathbf{r},\mathbf{r}')ds' + \frac{j}{\omega\epsilon}\int_{S'}(\nabla'\cdot\mathbf{J}_s)\nabla g(\mathbf{r},\mathbf{r}')dS' \tag{2.13}$$

which may be expressed more compactly as

$$\mathbf{E}^i_{\tan} = \int_{S'}\left[j\omega\mu\mathbf{J}_s g(\mathbf{r},\mathbf{r}') + \frac{j}{\omega\epsilon}(\nabla'\cdot\mathbf{J}_s)\nabla g(\mathbf{r},\mathbf{r}')\right]dS' . \tag{2.14}$$

Equation (2.14) is a form of the Electric Field Integral Equation (EFIE). The unknown quantity to be solved for is $\mathbf{J}_s$.

B. SOLUTION OF THE EFIE USING MM

The method of moments (MM) technique can be used to solve for $\mathbf{J}_s$ by expanding it into a series of basis functions, $\mathbf{J}_i$,

$$J_s = \sum_i^N C_i J_i \tag{2.15}$$

where the C_i are complex constants to be determined. Substitution of equation (2.15) into 2.14 gives

$$E^i_{tan} = \sum_{i=1}^N C_i \int_{S_i} \left[j\omega\mu J_i g(r,r') + \frac{j}{\omega\epsilon} (\nabla' \cdot J_i) \nabla g(r,r') \right] dS' \,. \tag{2.16}$$

To generate the required N equations to solve for the N unknowns, define a suitable weighting function $\mathbf{W}_k$, and take the inner product of $\mathbf{W}_k$ with both sides of equation 2.16. The inner product is defined such that it satisfies

$$\begin{aligned} &<w,v> = <v,w> \\ &<\alpha f + \gamma v, w> = \alpha <f,w> + \gamma <v,w> \\ &<v^*,v> > 0 \quad if\ v \neq 0 \\ &<v^*,v> = 0 \quad if\ v = 0 \end{aligned} \tag{2.17}$$

[Ref. 4]. Choose the following inner product:

$$<w,v> = \int_S w^* \cdot v dS \tag{2.18}$$

where * signifies the complex conjugate. This leads to

$$\int_{S_k} W_k \cdot E^i_{tan} dS = \int_{S_k} j\omega\mu W_k \cdot \sum_{i=1}^N C_i \int_{S_i} \left[J_i g(r,r') + \frac{j}{\omega\epsilon} (\nabla' \cdot J_i) \nabla g(r,r') \right] dS' dS \,. \tag{2.19}$$

Interchanging the order of summation and integration and applying the surface divergence theorem yields for the right hand side of equation (2.19)

$$\sum_{i=1}^{N} C_i \iint_{S_k S_i} \left[j\omega\mu(\boldsymbol{W}_k \cdot \boldsymbol{J}_i) g(\boldsymbol{r},\boldsymbol{r}') - \frac{j}{\omega\epsilon}(\nabla' \cdot \boldsymbol{J}_i)(\nabla \cdot \boldsymbol{W}_k) g(\boldsymbol{r},\boldsymbol{r}') \right] dS' dS \;. \tag{2.20}$$

By making the following definitions,

$$\begin{aligned} V_k &= \int_{S_k} \boldsymbol{W}_k \cdot \boldsymbol{E}^{i}_{\tan} dS \\ Z_{ik} &= \iint_{S_k S_i} \left[j\omega\mu(\boldsymbol{W}_k \cdot \boldsymbol{J}_i) g(\boldsymbol{r},\boldsymbol{r}') - \frac{j}{\omega\epsilon}(\nabla' \cdot \boldsymbol{J}_i)(\nabla \cdot \boldsymbol{W}_k) g(\boldsymbol{r},\boldsymbol{r}') \right] dS' dS \end{aligned} \tag{2.21}$$

2.20 and 2.21 can be written in matrix form,

$$[V] = [Z][C] \tag{2.22}$$

where [V], [Z] and [C] are called the generalized voltage, impedance and current matrices, respectively. The unknown [C] may be solved by an appropriate matrix inversion algorithm. Symbolically,

$$[C] = [Z]^{-1}[V] \;. \tag{2.23}$$

The generalized current matrix elements are the weighting coefficients in the summation of equation (2.15). The current J_s is computed from equation (2.15) and the scattered field is calculated using this current in the radiation integrals. It should be noted that [V], [Z], and [C] have units of volts, ohms, and amperes, but are not unique. In general, they depend on the choice of basis and weighting functions. However, the current will converge to the same numerical value as the number of basis functions are increased, provided the solutions are implemented correctly.

C. SPECIALIZATION OF THE EFIE TO A CIRCULAR LOOP USING CONFORMAL SUBSECTIONS.

The MM procedure will now be applied to a circular loop in the X-Y plane as illustrated in Figure 3. The loop is an ideal test geometry to study the characteristics of a MM solution using conformal subsections. It is a relatively simple geometry and other accurate solution methods are available to evaluate the performance of conformal subsections. The loop has a radius r_0 and is divided into N conformal segments. In this case a conformal segment is a circular arc. The arclength of the i^{th} segment is,

$$\Delta l_i = l_{i+1} - l_i = r_0 \Delta \phi_i \ . \tag{2.24}$$

The basis functions, J_i, of equation (2.15) are chosen to be overlapping triangles,

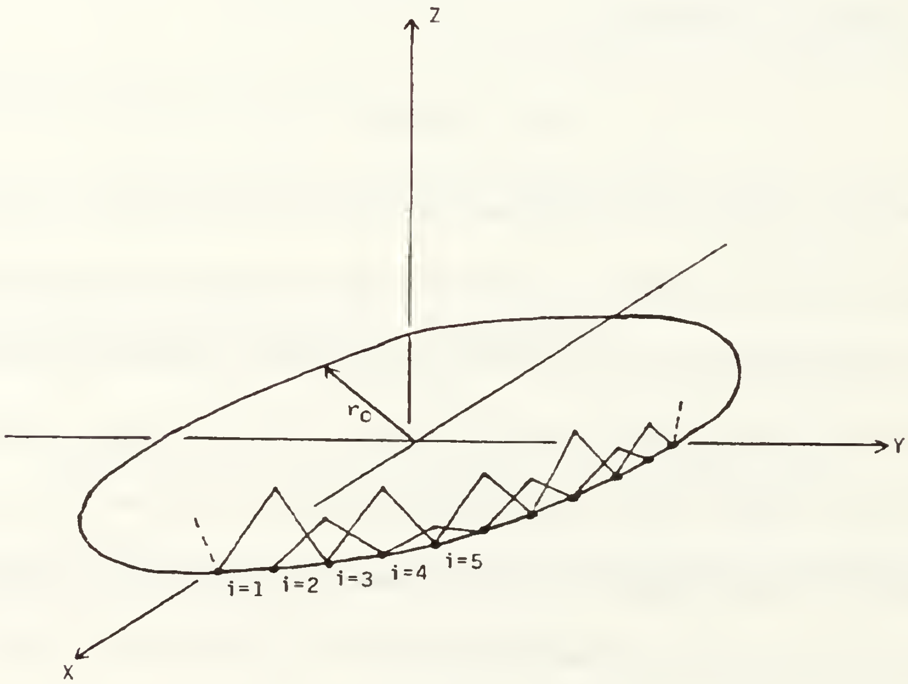


Figure 3. Thin Wire Loop Geometry

$$J_i = T_i(l)\hat{l} = \begin{cases} \dfrac{\hat{l}(l-l_i)}{2\pi a \Delta l_i}; & l_i < l \le l_{i+1} \\ \dfrac{\hat{l}}{2\pi a}\left(1 - \dfrac{l-l_{i+1}}{\Delta l_{i+1}}\right); & l_{i+1} < l \le l_{i+2} \\ 0; & elsewhere \end{cases} \quad . \tag{2.25}$$

Triangular basis functions are chosen because they are a more accurate representation of the current than a pulse basis function since the current is continuous everywhere along the wire, and they are relatively easy to deal with analytically. Balanis [Ref. 5] states that increasing the basis function complexity beyond triangles may not be warranted by the additional improvement in convergence rate. The triangular basis functions span two segments and overlap as shown in Figure 3. Therefore, the resultant current will be piecewise linear. The weighting (or testing) functions $\mathbf{W}_k$ in equation (2.19) are chosen such that $\mathbf{W}_k=\mathbf{J}_k$ (Galerkin's method). Wang [Ref. 6] states that Galerkin's method provides numerical results which are more accurate than other testing methods under similar computational constraints. Substitution of the above weighting and basis functions in the expression for Z_{ik} in equation (2.21) yields

$$Z_{ik} = \int_{S_k}\int_{S_i}\left[j\omega\mu T_k(l)T_i(l')(\hat{l}\cdot\hat{l}') - \frac{j}{\omega\epsilon}T'_i(l')T'_k(l)\right]g(r,r')\,dS'dS \tag{2.26}$$
$$where\ T'_i(l') = \frac{\partial T_i}{\partial l'}, \quad T'_k(l) = \frac{\partial T_k}{\partial l} \ .$$

Using the following relations,

$$
\begin{aligned}
\hat{l}\cdot\hat{l}' &= \hat{\phi}\cdot\hat{\phi}' = \cos(\phi-\phi') \\
l &= r_0\phi; \quad l' = r_0\phi' \\
dS &= 2\pi a r_0 d\phi; \quad dS' = 2\pi a r_0 d\phi' \\
T_i(\phi) &= T_i(r_0\phi) = T_i(l) \\
T'_i(l') &= \frac{1}{r_0}T'_i(\phi) \\
\eta &= \sqrt{\frac{\mu}{\epsilon}}; \quad \beta = \omega\sqrt{\mu\epsilon} \\
g(\boldsymbol{r},\boldsymbol{r}') &= \frac{e^{-j\beta|\boldsymbol{R}|}}{4\pi|\boldsymbol{R}|}; \quad \boldsymbol{R} = \boldsymbol{r}-\boldsymbol{r}'
\end{aligned}
\tag{2.27}
$$

equation (2.26) may be written as

$$
Z_{ik} = \frac{r_0^2 j\beta\eta}{4\pi}\int_{\phi_k}^{\phi_{k+2}}\int_{\phi_i}^{\phi_{i+2}}\left[T_k(\phi)T_i(\phi')\cos(\phi-\phi') - \frac{1}{\beta^2 r_o^2}T_i'(\phi')T_k'(\phi)\right]\frac{e^{-j\beta|\boldsymbol{R}|}}{|\boldsymbol{R}|}d\phi\, d\phi' \tag{2.28}
$$

From Figure 4 and the law of cosines, $|\mathbf{R}|$ is given by,

$$
|\boldsymbol{R}|^2 = 2r_0^2[1-\cos(\phi-\phi')]+a^2 \tag{2.29}
$$

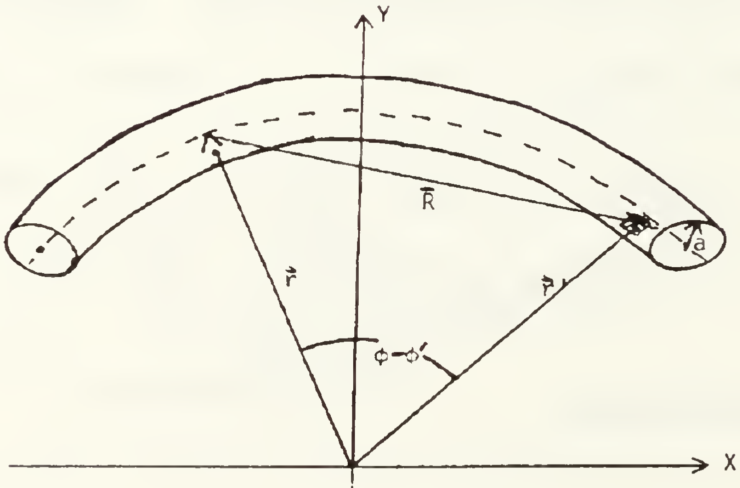


Figure 4. Geometry for Determining R.

and using the trigonometric identity,

$$\sin^2\left(\frac{\alpha}{2}\right) = \frac{1}{2}(1-\cos\alpha) \tag{2.30}$$

results in,

$$|R| = r_0\sqrt{4\sin^2\left(\frac{\phi-\phi'}{2}\right)+\frac{a^2}{r_0^2}} \ . \tag{2.31}$$

By choosing the test (unprimed) points at the center of the wire and the source points on the surface of the wire, the singularities along the diagonal of [Z] at $\phi=\phi'$ where $\mathbf{r}=\mathbf{r}'$ are avoided. The technique used to calculate $|\mathbf{R}|$ is discussed further in Chapters III and IV.

The voltage elements, V_k, given in equation (2.21) become

$$V_k = r_0 \int_{r_0\phi_k}^{r_0\phi_{k-2}} T_k(r_0\phi)\cdot E^i d\phi \ . \tag{2.32}$$

The incident field, $\mathbf{E}^i$, for the purpose of this study, will be a plane wave. Figure 5 shows the direction of incidence of the plane wave in spherical polar angles $\theta=\Theta$ and $\phi=\Phi$ measured from the Z and X axes, respectively. $\mathbf{E}^i$ can be θ or ϕ polarized. Referring to Figure 5, for θ polarization, the component of $\mathbf{E}^i$ tangential to the loop, is

$$E_\theta^i\hat{\Theta}\cdot\hat{\phi} = E_\theta^i\cos\Theta\,\sin(\Phi-\phi) \ . \tag{2.33}$$

Similarly, for ϕ polarization,

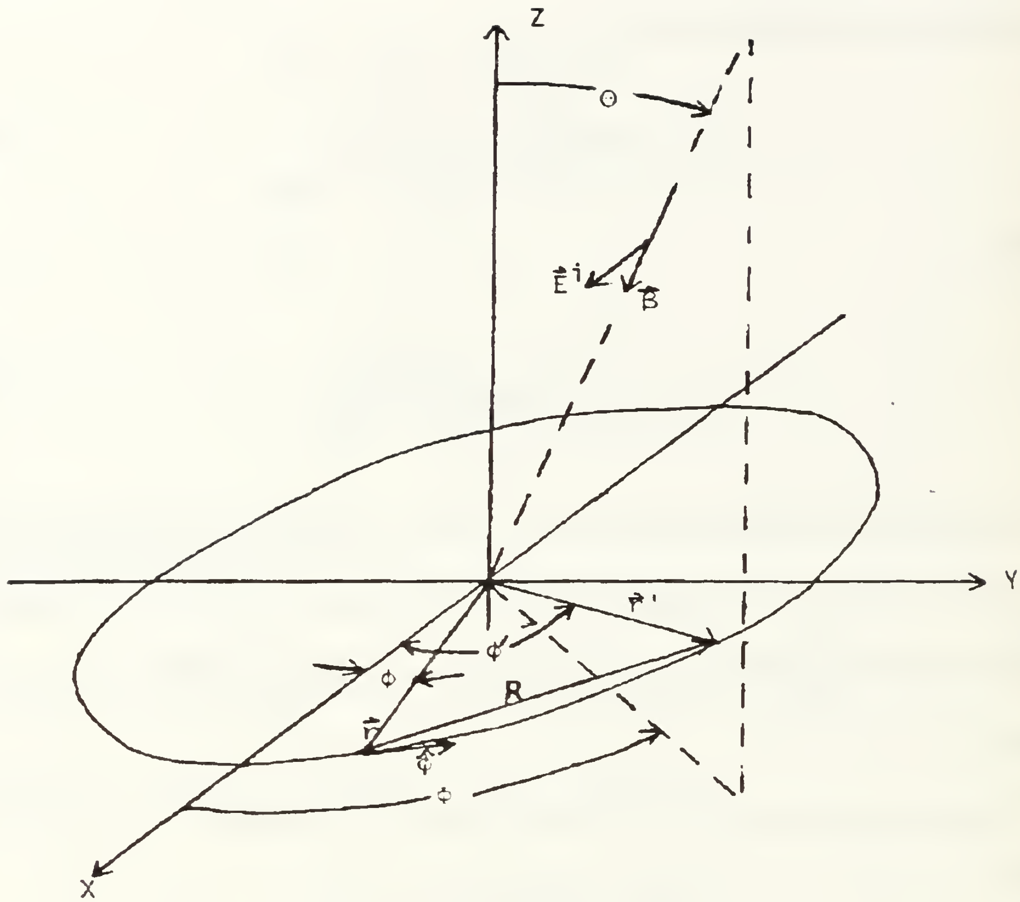


Figure 5. Plane Wave Incident on Circular Loop

$$E^i_\phi \hat{\Phi} \cdot \hat{\phi} = E^i_\phi \cos(\Phi - \phi) \ . \tag{2.34}$$

The component of the phase vector, β, parallel to **r** is

$$\hat{\beta} \cdot \hat{r} = \sin\Theta \ \cos(\Phi - \phi) \ . \tag{2.35}$$

Equations (2.30) and (2.32) combine with 2.29 to give

$$V_{\theta k} = r_0 E^i_\theta \cos\Theta \int_{\phi_k}^{\phi_{k+2}} T_k(\phi) \sin(\Phi - \phi) e^{-j\beta r_0 \sin\Theta \ \cos(\Phi - \phi)} d\phi \ . \tag{2.36}$$

A similar expression for a ϕ directed incident field is

$$V_{\phi k} = r_0 E_\phi^i \int_{\phi_k}^{\phi_{k+2}} T_k(\phi) \cos(\Phi - \phi) e^{-j\beta r_0 \sin\Theta \cos(\Phi - \phi)} d\phi \ . \tag{2.37}$$

The computer coding of the solution for the thin wire loop using the equations developed above is described in the next chapter.

III. COMPUTER CODES FOR THE THIN WIRE LOOP

In this section, the FORTRAN program for a thin wire loop using curved segments is discussed. The results are presented and compared to similar solutions using straight subsections and Fourier modes.

A. DESCRIPTION OF THE CODES

Table 1 summarizes the organization of the three programs. Computer listings are given in Appendix B and equations from Chapter II will be referenced with a "2." preceding the equation number. The FORTRAN source code for the conformal subsections is named CURVENEW, and the codes for the straight subsections and the Fourier mode solution are named LOOPSCAT and HARLOOP respectively.

CURVENEW, LOOPSCAT and HARLOOP are functionally similar. Each calculates the loop geometry based on the segment size, loop radius, and wire radius, and each uses Gaussian quadrature for numerical integration. CURVENEW computes the impedance matrix, [Z], in subroutine ZCURVED from the loop geometry of Figure 5 using equation (2.28). LOOPSCAT uses a similar formulation applied to straight segments in subroutine ZMATWW. HARLOOP computes [Z] using the equations developed in reference [7]. CURVENEW computes the excitation vector, [V], using equation (2.36) or (2.37) in subroutine CURVEW. LOOPSCAT uses a similar formulation for straight subsections in subroutine PLANEW. HARLOOP computes [V]

using the equations in reference [7] in subroutine PLANEW. In CURVENEW, all integrals are evaluated numerically and symmetry of the impedance elements is used to fill the [Z] matrix and reduce the number of numerical calculations. Two formulations were investigated with LOOPSCAT: One using a delta function approximation to evaluate one of the double integrals in equation (2.21) and the other using Gaussian quadrature for both integrations. CURVENEW does all numerical integrations using Gaussian quadrature. Matrix symmetry is also used in LOOPSCAT to reduce the number

TABLE 1. FUNCTIONAL SUMMARY OF PROGRAMS

Program--> Function	CURVENEW (Curved Segments)	LOOPSCAT (Straight Segments)	HARLOOP (Fourier modes)
Read Input Parameters	Lines 25-38	Lines 15-27	Lines 19-30
Establish Loop Geometry	Lines 45-71	Lines 53-76	Lines 44-54
Calculate [Z]	Subroutine **ZCURVED**	Subroutine **ZMATWW**	Subroutine **ZMATWW**
Calculate [V]	Subroutine **CURVEW**	Subroutine **PLANEW**	Subroutine **PLANEW**
Solve System $[C]=[Z]^{-1}[V]$	Subroutines **DECOMP** and **SOLVE**	Subroutines **DECOMP** and **SOLVE**	Subroutines **DECOMP** and **SOLVE**
Calculate E^S	Lines 140-180	Lines 160-197	Lines 93-137

of calculations for [Z]. Computation of [C] is performed in subroutines DECOMP and SOLVE [Ref. 3], which solve the system of equations using Gaussian elimination. Subroutines DECOMP and SOLVE are common to all three programs. The subroutines ZCURVED and CURVEW are discussed in more detail in the next section.

1. Loop Geometry

To generate the loop geometry an initial estimate of the desired circular arc length, Δl, is provided. This is used to estimate an angular increment, $\Delta\phi$,

$$\Delta\phi = \frac{\Delta l}{r_0} . \tag{3.1}$$

From this estimate, the number of generating points is calculated by,

$$N = Int\left(\frac{2\pi}{\Delta\phi}\right) + 1 \tag{3.2}$$

and $\Delta\phi$ is recalculated using,

$$\Delta\phi = \frac{2\pi}{N} \tag{3.3}$$

to ensure that $\Delta\phi$ is such that N segments fill exactly 2π radians. The loop points P_1 and P_2 are coincident with P_{N-1} and P_{N-2} so that the current is continuous around the loop.

2. Subroutines ZCURVED and ZMATWW

Subroutines ZCURVED and ZMATWW take advantage of the symmetry that exists on the loop. For all basis functions, the self impedance terms are equal. The

remainder of the matrix is filled with impedance values that repeat in a cyclic manner. Mathematically,

$$
\begin{gathered}
Z_{11} = Z_{22} = Z_{33} = \ldots = Z_{NN} \\
Z_{12} = Z_{23} = Z_{34} = \ldots = Z_{N-1\,N} \\
Z_{13} = Z_{24} = Z_{35} = \ldots = Z_{N-2\,N} \\
\vdots \\
Z_{1\,N-1} = Z_{2\,N}
\end{gathered}
\tag{3.4}
$$

The elements along any diagonal of [Z] are equal and the lower off-diagonal elements are the mirror image of the upper diagonals. Thus [Z] is a symmetrical Toeplitz matrix. Therefore, computation of the first row of [Z] provides enough information to fill the entire matrix.

Because of the Green's function in the integrand for the impedance elements, the numerical treatment of the self term is very important. To optimize the convergence rate and accuracy of CURVENEW and LOOPSCAT, several different approaches are used to evaluate $|\mathbf{R}|$ near the singularity point where $\mathbf{r}=\mathbf{r}'$. In the first method, the observation point is chosen along the axis of the wire and the source point along the surface for all i,k, giving $|\mathbf{R}|=a$ at $\phi=\phi'$ (equation (2.31)). For the second method, both the observation point and the source point are chosen along the axis of the wire except on the segment $i=k$ where the value of ϕ at the midpoint is chosen on the axis of the wire, with $\mathbf{r}'$ on the surface of the wire. Finally, both the observation point and the source point are chosen along the axis of the wire, except on the segment $i=k$, where $\mathbf{r}$ is chosen along the axis, and $\mathbf{r}'$ is chosen on the surface. Choosing the source point and

observation point as in the first case gives the most accurate results, but only slightly more accurate than the third case. Case two is accurate for small segment sizes but is inaccurate for larger segment sizes. Case three was selected for both CURVENEW and LOOPSCAT because it is only slightly less accurate than case one, and required fewer lines of code.

ZCURVED calculates the impedance elements of the first row of [Z] by breaking the integral in equation (2.28) into four parts. For example, in the first row of [Z], Z_{1i}, is calculated by summing contributions from the following four regions of integration (Figure 6):

1. A double integration along the positive slope of the T_1 over segment 1 and the positive slope of T_i over segment i.
2. A double integration along the negative slope of the T_1 over segment 2, and the positive slope of T_i over segment i.
3. A double integration along the positive slope of the T_1 over segment 1, and the negative slope of T_i over segment i+1.
4. A double integration along the negative slope of the T_1 over segment 2, and the negative slope of T_i over segment i+1.

The integrations are computed in a similar manner for the derivatives of the basis functions, T_1' and T_i', over the same subsections. A similar procedure is used for the straight subsections in subroutine ZMATWW to calculate the impedance elements.

The numerical integrations are performed using Gaussian quadrature, with the number of points per subsection specified as an input parameter to program GAUSWGT, which computes the Legendre polynomial coefficients for a specified

number of integration points and writes them to file OUTGLEG. Gaussian quadrature was chosen because it requires fewer function evaluations than other methods for a given accuracy and does not require equal interval samples [Ref. 8]. CURVENEW, LOOPSCAT and HARLOOP read the coefficients from file OUTGLEG. The number of integration points per wavelength was varied to optimize convergence of LOOPSCAT and CURVENEW and is discussed in more detail in Chapter IV. The excitation vector [V] is calculated from equation (2.36) or (2.37) in subroutine CURVEW of CURVENEW.

3. Execution Time

Analysis of the nested DO loop structure of subroutine ZCURVED of program CURVENEW indicates that the total execution time of ZCURVED can be represented by,

$$T_{1c} = \alpha_c N_c N_{gc}^2 \tag{3.5}$$

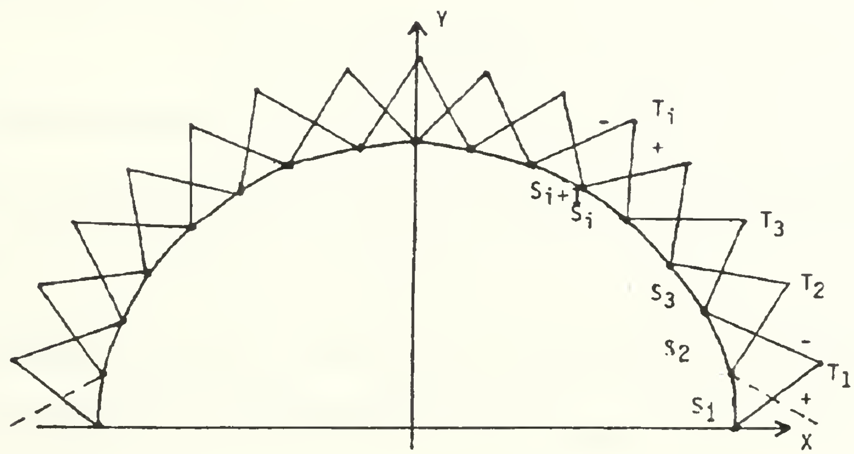


Figure 6. Loop Geometry for Impedance Integrations

where N_c is the number of curved segments (from equation (3.2)) N_{gc} is the number of Gaussian integration constants per curved segment and α_c is a constant. Execution time of the subroutines DECOMP and SOLVE, common to both CURVENEW and LOOPSCAT, can be represented similarly by [Ref. 9],

$$T_2 = \gamma N^3$$

where N is the number of segments. The excitation subroutine CURVEW execution time and field integrations are given by,

$$T_{3c} = \zeta_c N_c N_{gc} \tag{3.7}$$

where again γ and ξ_c are constants. Assuming that the execution time of the rest of the program is negligible, the total execution time of CURVENEW is

$$T_c = T_{1c} + T_2 + T_{3c} = \alpha_c N_c N_{gc}^2 + \gamma N_c^3 + \zeta_c N_c N_{gc} \ . \tag{3.8}$$

A similar expression for LOOPSCAT uses the subscript l,

$$T_l = \alpha_l N_l N_{gl}^2 + \gamma N_l^3 + \zeta_l N_l N_{gl} \ . \tag{3.9}$$

Run times were recorded for various values of N_c, N_l and N_g using an IBM PC/AT with a math coprocessor and the coefficients for CURVENEW are found to be γ=0.000156, α_c=0.0230, and ξ_c=0.0222. The coefficients for LOOPSCAT are α_l=0.0132 and ξ_l=0.0252. For the moment, assume that the number of Gaussian integration points per wavelength, N_g, is held constant for both CURVENEW and LOOPSCAT. The number of integration points on a segment is,

$$N_{gc} = N_g \Delta l_c \ , \tag{3.10}$$

and the number of segments is,

$$N_c = \frac{2\pi r_0}{\Delta l_c} \ . \tag{3.11}$$

Similar expressions may be written for straight subsections. Combining equations (3.8), 3.10, and 3.11 gives

$$T_c = 4\pi^2 r_0^2 \frac{N_g^2 \alpha_c}{N_c} + \gamma N_c^3 + 2\pi r_0 \zeta_c N_g \ . \tag{3.12}$$

The ratio of T_c to T_l is given by,

$$T_c / T_l = \frac{4\pi^2 r_0^2 N_g^2 \alpha_c / N_c + \gamma N_c^3 + 2\pi r_0 \zeta_c N_g}{4\pi^2 r_0^2 N_g^2 \alpha_l / N_l + \gamma N_l^3 + 2\pi r_0 \zeta_l N_g} \ . \tag{3.13}$$

Equation (3.13) will be used in Chapter IV to compare the execution times of CURVENEW and LOOPSCAT.

IV. CALCULATED DATA FOR THE LOOP

The convergence of the MM solutions for both the current and electric field for circular loops of various dimensions are presented for both linear and circular polarizations. The convergence is shown to depend on the segment size and number of integration points, as well as excitation conditions (incidence direction and polarization). Representative plots are presented within the chapter, and additional plots are given in Appendix B.

A. CONVERGENCE OF HARLOOP

The Fourier mode solution, HARLOOP, was tested for convergence with respect to incidence angle, number of modes, and number of integration constants to establish a baseline for comparison to CURVENEW and LOOPSCAT. HARLOOP was chosen as a baseline because the sinusoidal basis functions match the physical behavior of the current on the loop, and thus the current series converges rapidly. This is illustrated in Figure 7 for a 0.5 λ radius loop with a plane wave incident at an angle of 40 degrees.

Oscillation of the current as a function of ϕ becomes more rapid as Θ is increased, because the phase of the incident field over the loop varies as sin Θ (equation (2.35)). For a θ polarized linear wave incident in the $\phi=0$ plane, the current is always zero at $\phi=0$ and 180 degrees, where $\mathbf{E}^i$ is cross-polarized with respect to the axis of the wire. For θ polarized incident waves, maxima occur at $\phi=90$ and 270 degrees, where $\mathbf{E}^i$ is

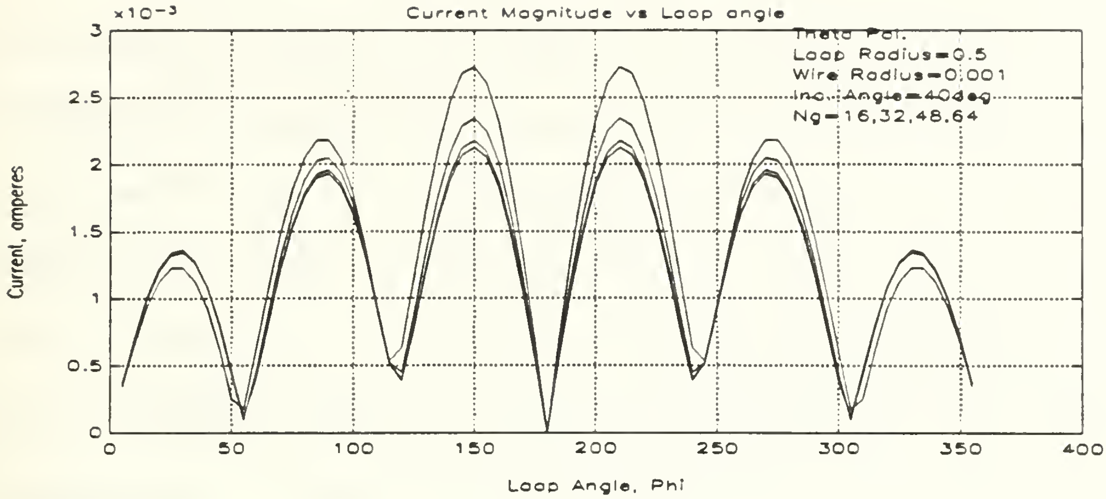


Figure 7. Convergence of the Current in the Complex Exponential Solution

parallel to the axis of the wire. The overall current amplitude decreases for Θ approaching 90 degrees, as expected, since the loop's projected area is small as viewed by the incident wave. For ϕ polarized incident waves, the minima occur at $\phi=90$ and 270 degrees and maxima at $\phi=0$ and 180 degrees. The currents do not vanish for Θ approaching 90 degrees because the loop is parallel to the ϕ polarized incident field. Circularly polarized incident waves give a constant magnitude current at normal incidence, and oscillations increase with Θ. For $\Theta=90$ degrees, the circular and ϕ polarization responses are identical.

HARLOOP is also found to be in agreement with measurements taken on the echo area of wire loops at normal incidence [Ref. 10]. The plot of Figure 8 gives the echo area (σ/λ^2) versus r_0 for varying wire radius using HARLOOP. Measured data is indicated by the '+' sign.

B. CONVERGENCE OF THE CURRENT EXPANSION

Having established HARLOOP as a baseline for comparison, convergence of the curved subsection program CURVENEW and the linear subsection program LOOPSCAT was evaluated. The plots of Figures 9 through 14 give the current on the loop as a function of loop angle, ϕ, for varying Θ, loop radius, and incident wave polarization. The number of integration points per wavelength, N_g, is held constant at 320 in LOOPSCAT and CURVENEW. This number was chosen empirically to give a converged current within five to ten percent RMS. The RMS error is defined relative to the Fourier mode solution. The HARLOOP current is plotted with the solid line, those of CURVENEW are plotted with the "+" sign, and those of LOOPSCAT with the "o". The wire radius is 0.001 λ for these calculations.

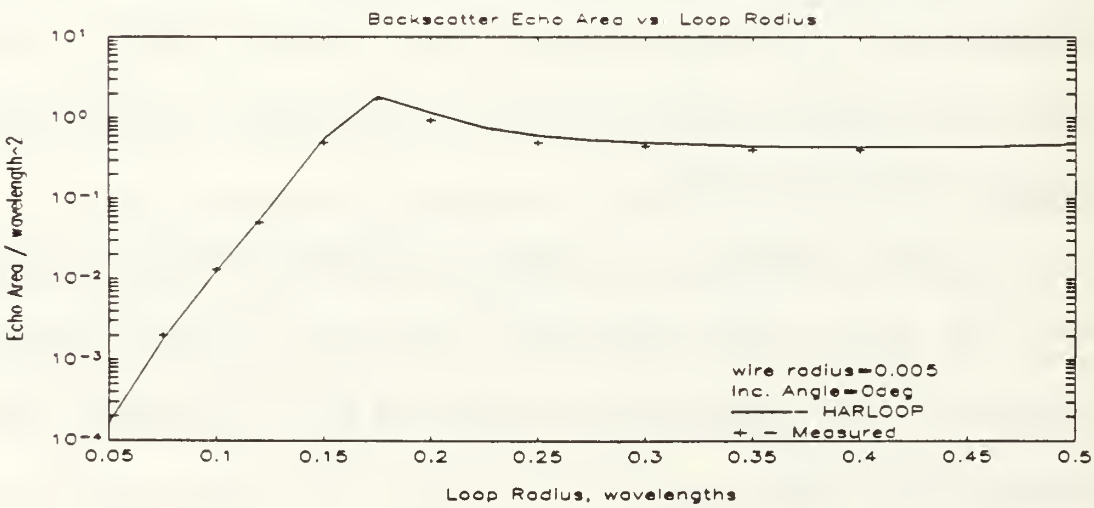


Figure 8. Backscatter Echo Area for a Loop with varying Radius at Normal Incidence

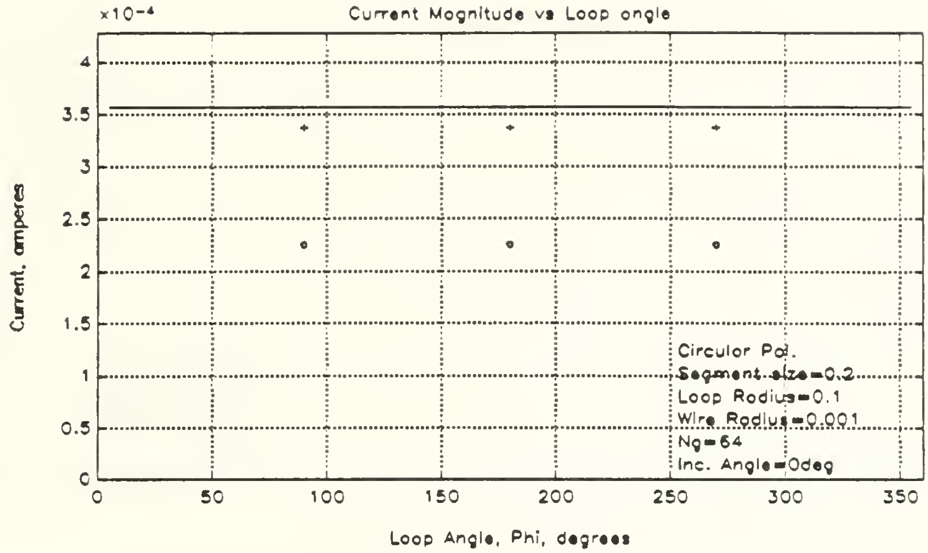


Figure 9. Magnitude of the Current on a 0.1 λ Radius Loop, Normal Incidence, Circular Polarization (+ =curved; o =linear)

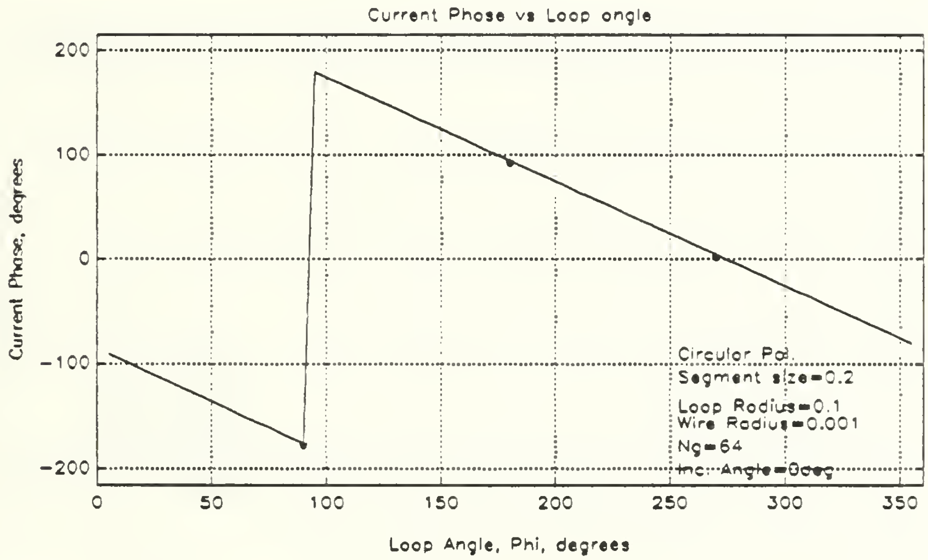


Figure 10. Phase of the Current on a 0.1 λ Radius Loop, Normal Incidence, Circular Polarization (+ =curved; o =linear)

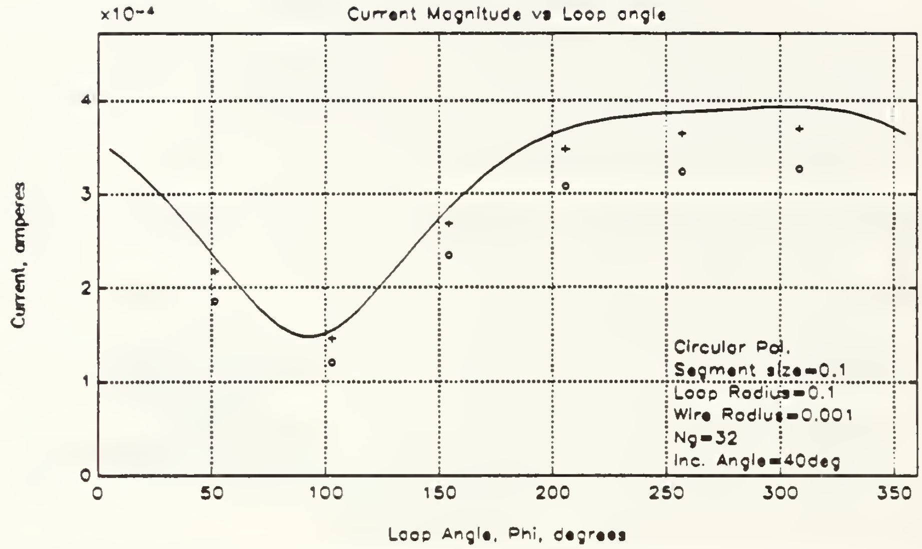


Figure 11. Magnitude of the Current on a 0.1 λ Radius Loop, Incidence Angle=40 deg, Circular Polarization (+ =curved; o =linear)

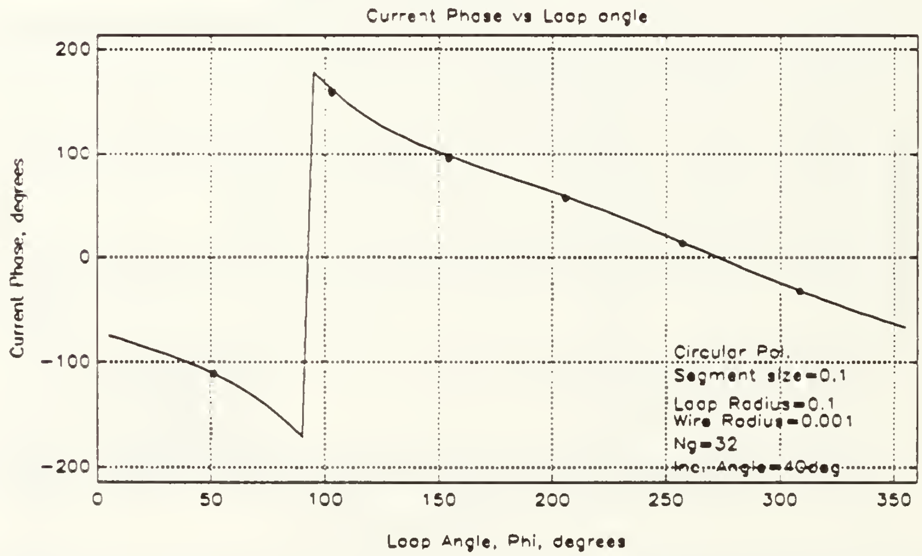


Figure 12. Phase of the Current on a 0.1 λ Radius Loop, Incidence Angle=40 deg., Circular Polarization (+ =curved; o =linear)

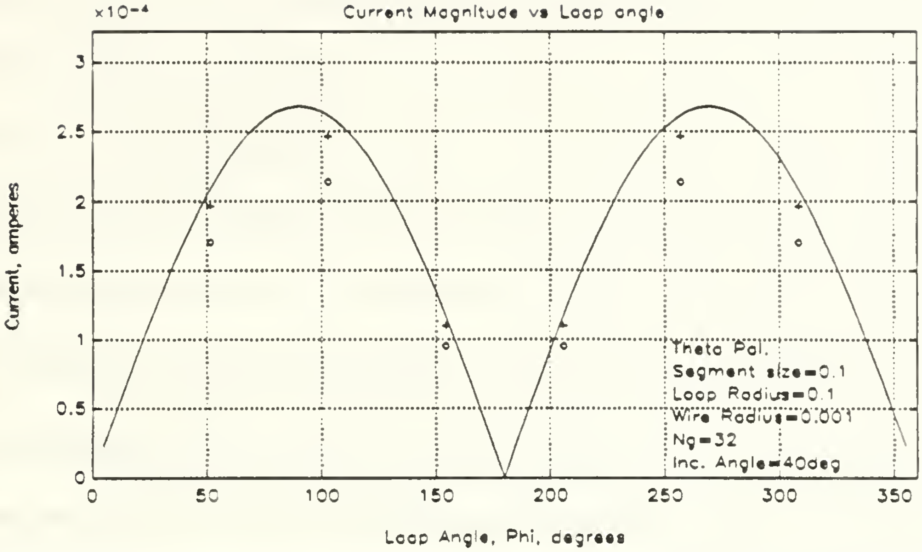


Figure 13. Magnitude of the Current on a 0.1 λ Radius Loop, Incidence Angle=40 deg., Theta Polarization (+ =curved; o =linear)

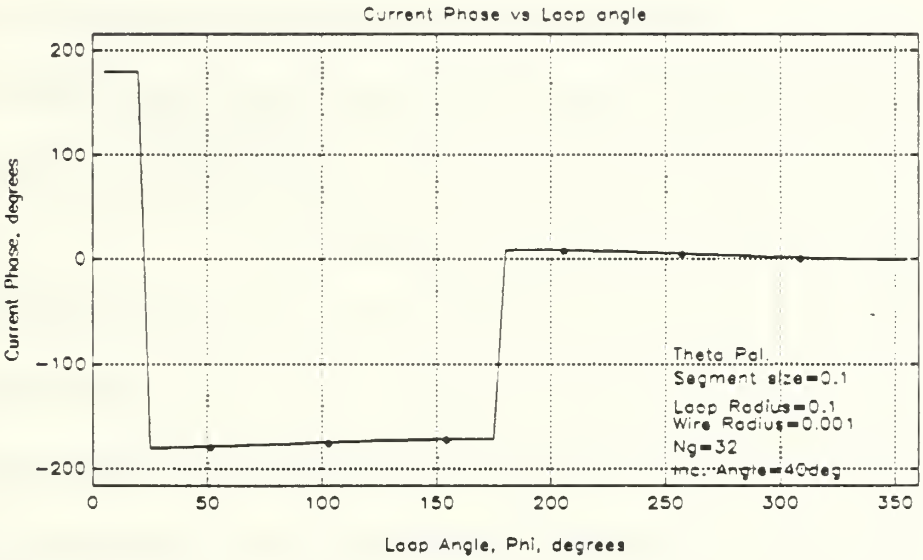


Figure 14. Phase of the Current on a 0.5 λ Radius Loop, Incidence Angle=40 deg., Theta Polarization (+ =curved; o =linear)

Representative plots of the normalized root mean squared error are given in Figures 15 through 22. For set values of N and N_g, CURVENEW converges faster than LOOPSCAT in most cases. The error difference is most pronounced for loop circumferences on the order of a wavelength or less. From the plots, for r_o=0.1 λ, CURVENEW converges to less than 10 percent error for segment sizes ranging from 0.02 to 0.2 wavelengths (N_c=32 to N_c=3). LOOPSCAT converges to within 10 percent error for segment sizes less than approximately 0.06 λ (N_l > 10) but gives errors of 30 to 40 percent for a segment size of 0.2 wavelengths. There is no improvement using curved subsections on larger loops for off-axis incidence waves, but the curved subsections give small improvements for large loops at normal incidence. This is expected in view of the behavior of the current on the loop. For linear polarization the current abruptly flips polarity from one side of the loop to the other. For circular polarization, the amplitude is constant, but the phase is linear. Both of these conditions can be represented accurately by a few triangles if the impedance and excitation integrals are evaluated precisely on the loop contour (see Figure 23).

Plots of the magnitude of the backscattered $\mathbf{E}^s$ versus incidence angle for varying segment size, N_g, and r_0 are given in Figures 24 and 25. As with the currents, the backscattered field converges more rapidly for CURVENEW than LOOPSCAT in most cases, with the greatest difference for small radius loops and angles near the maximum values of $|\mathbf{E}^s|$.

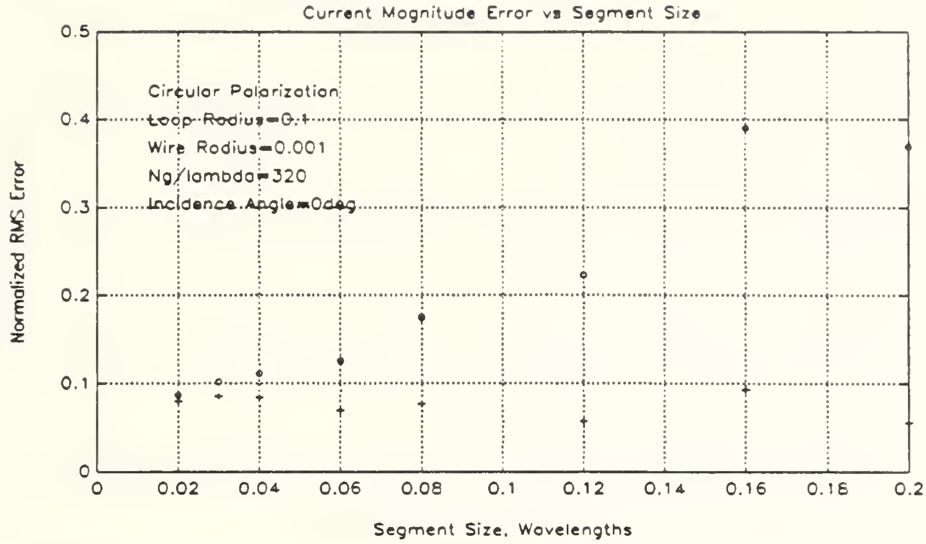


Figure 15. Error in the Current Magnitude for a 0.1 λ Radius Loop, Normal Incidence, Circular Polarization (+ =curved; o =linear)

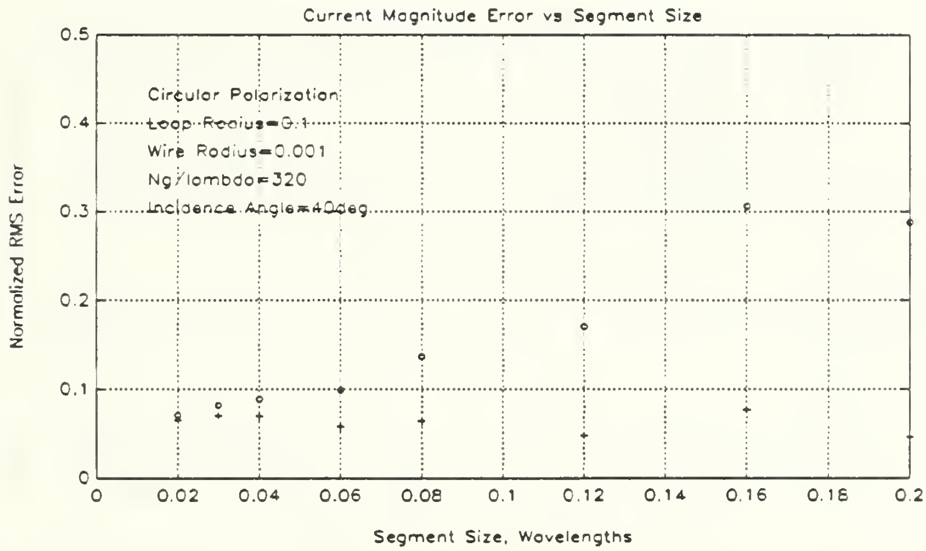


Figure 16. Error in the Current Magnitude for a 0.1 λ Radius Loop, Incidence Angle=40 deg., Circular Polarization (+ =curved; o =linear)

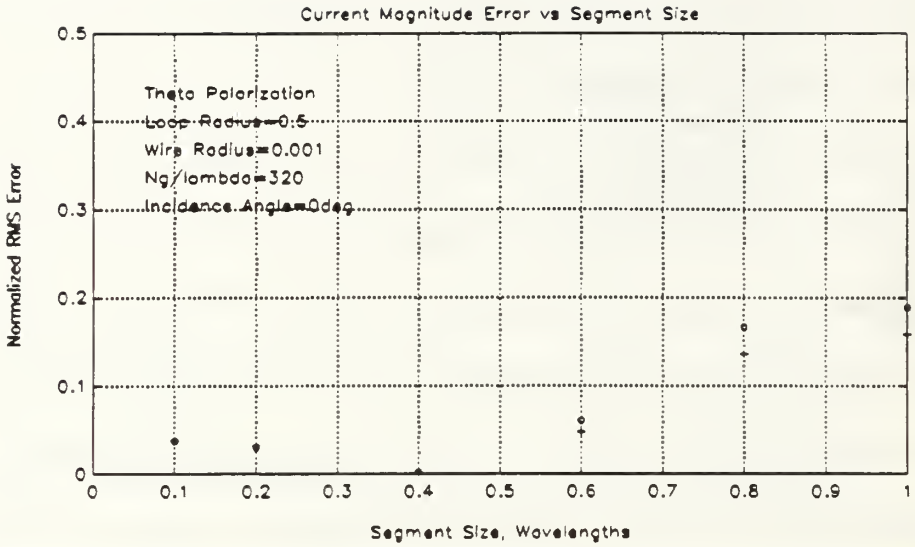


Figure 17. Error in the Current Magnitude for a 0.5 λ Radius Loop, Normal Incidence, Circular Polarization (+ =curved; o =linear)

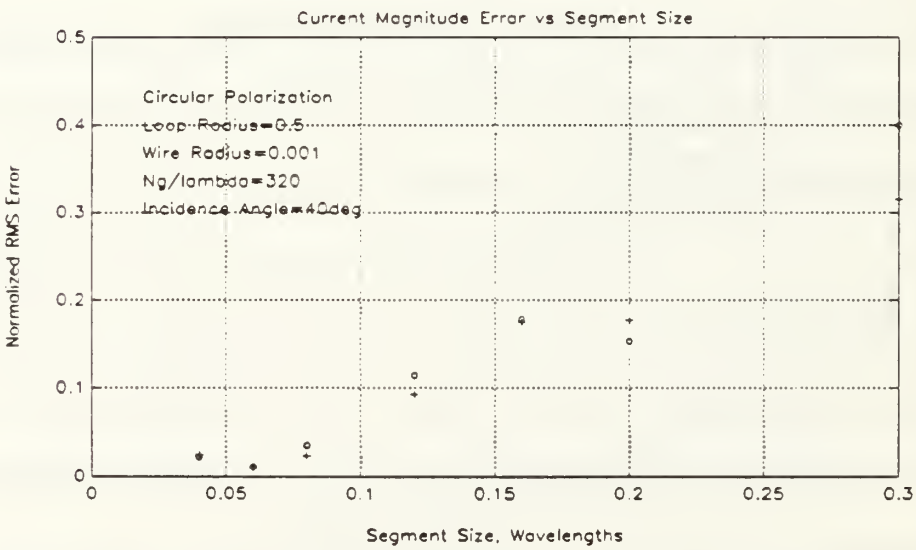


Figure 18. Error in the Current Magnitude for a 0.5 λ Radius Loop, Incidence Angle=40 deg., Circular Polarization (+ =curved; o =linear)

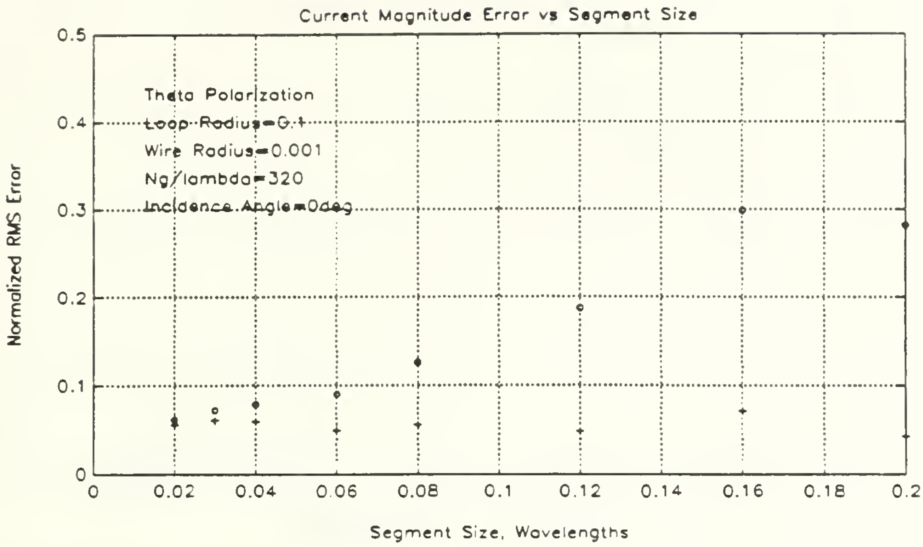


Figure 19. Error in the Current Magnitude for a 0.1 λ Radius Loop, Normal Incidence, Theta Polarization (+ =curved; o =linear)

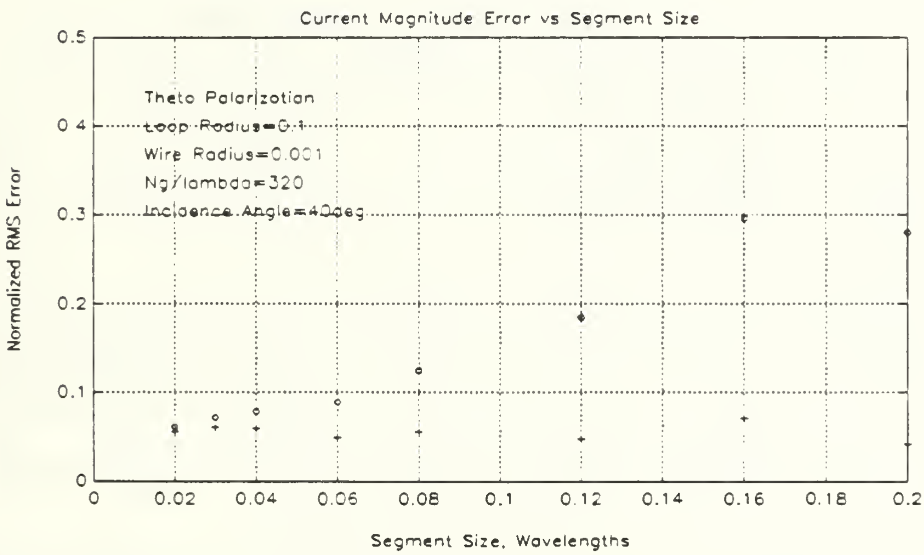


Figure 20. Error in the Current Magnitude for a 0.1 λ Radius Loop, Angle of Incidence=40 deg., Theta Polarization (+ =curved; o =linear)

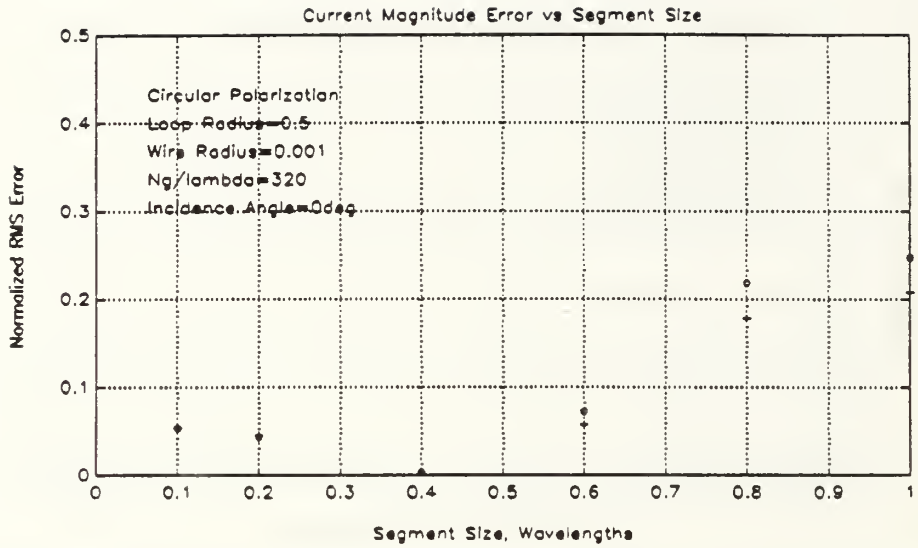


Figure 21. Error in the Current Magnitude for a 0.5 λ Radius Loop, Normal Incidence, Theta Polarization (+ =curved; o =linear)

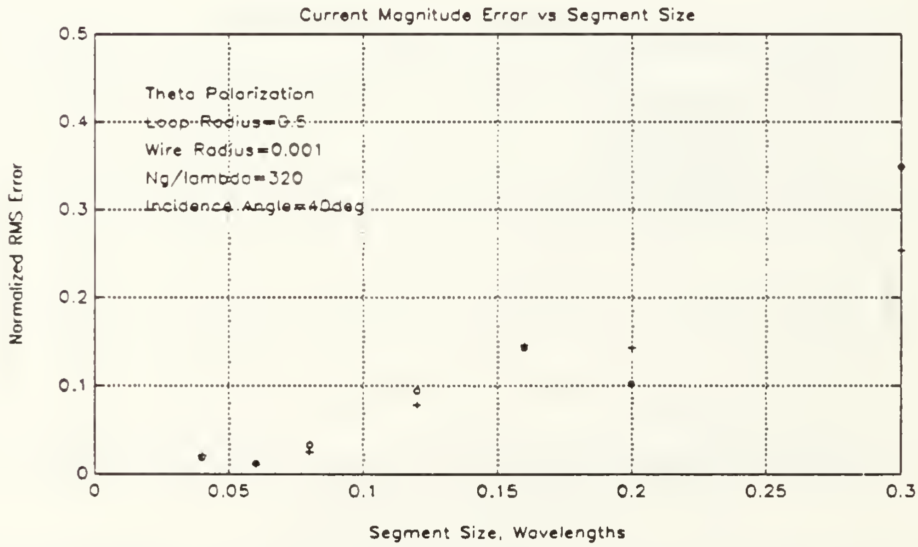


Figure 22. Error in the Current Magnitude for a 0.5 λ Radius Loop, Incidence Angle=40 deg., Theta Polarization (+ =curved; o =linear)

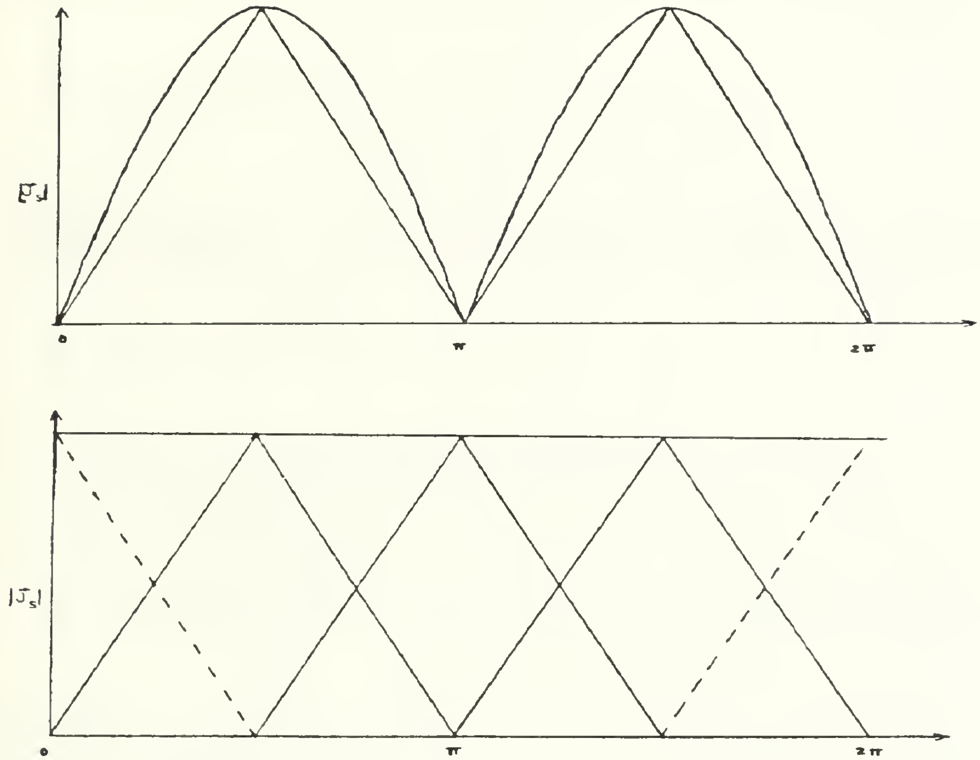


Figure 23. A Representation of a Sinusoidal Current with Two Basis Functions (top) and a Constant Current as a Superposition of Triangles (bottom)

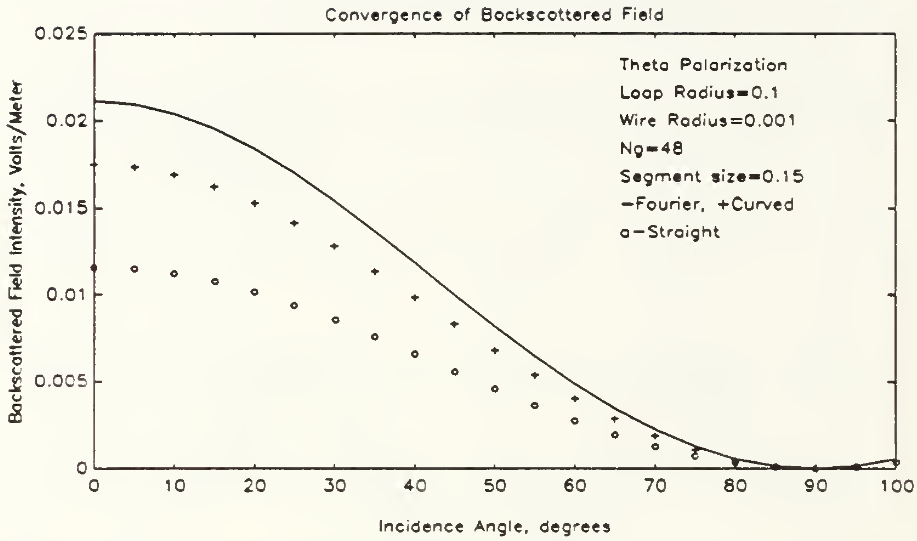


Figure 24. Backscattered Electric Field Intensity for varying Angles of Incidence, 0.1 λ Radius Loop, Theta Polarization (+ =curved; o =linear)

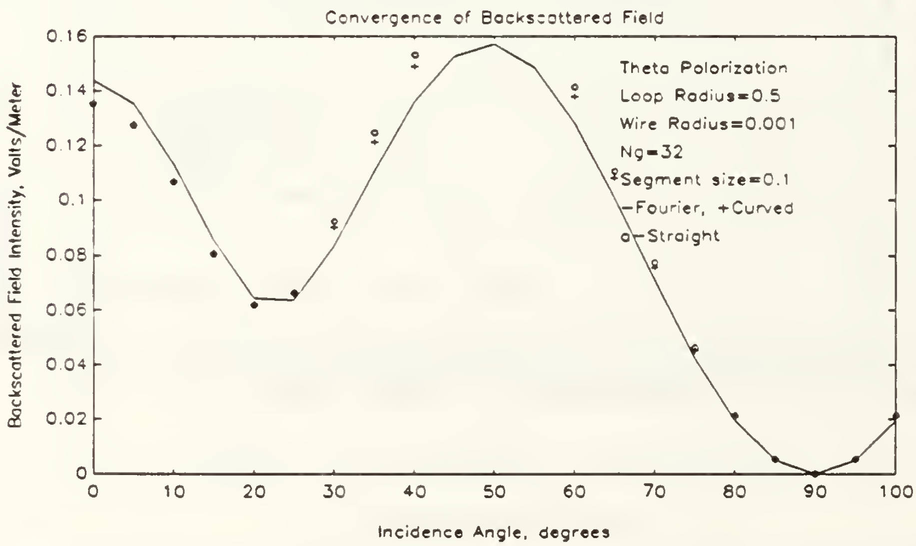


Figure 25. Backscattered Electric Field Intensity for varying Angles of Incidence, 0.5 λ Radius Loop, Theta Polarization (+ =curved; o =linear)

C. COMPARISON OF EXECUTION TIME AND MEMORY REQUIREMENTS

The average run time for a given loop radius and convergence error is greater for CURVENEW than for LOOPSCAT due to the N_g^2 dependence in equation (3.13) and relative magnitudes of the coefficients α and γ. The plot of equation (3.13) in Figure 26 illustrates the ratio of run times of CURVENEW and HARLOOP versus N_l for N_c=4,

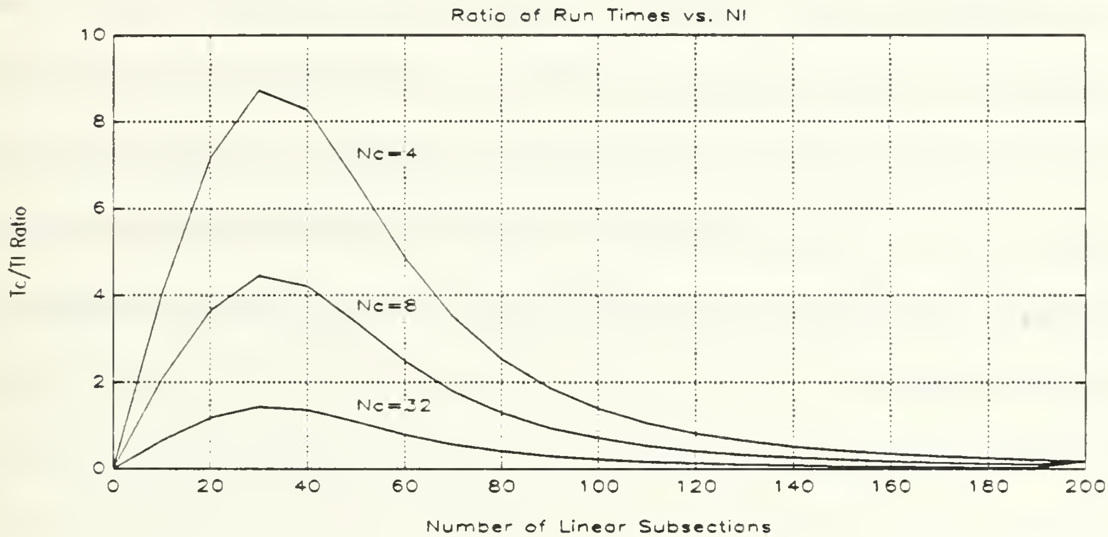


Figure 26. Ratio of Execution Times for CURVENEW and LOOPSCAT for Fixed Curved Segment Lengths and with Varying Linear Segment Lengths

8, 32 and N_g=320 for a 0.1 λ loop. The run time of CURVENEW is less than HARLOOP for $T_c/T_l < 1$. From the plot, the "break even" points are approximately N_l=115, 90, 52 for N_c=4, 8, 32. For N_l less than these values, the integration subroutine ZCURVED is the determining factor in the run time; for N_l greater than these values, Gausssian elimination subroutines DECOMP and SOLVE are the determining factors. The increased number of integrations per segment completely offset the time

savings of a reduced [Z] matrix in CURVENEW. As mentioned in Chapter III, a delta function approximation for the outer integration of the impedance integral of equation (2.28) was investigated. This reduces the exponent of N_g to one in equation (3.13), but many more segments are required for a given accuracy. A more efficient integration scheme using a large number of integration points in the vicinity of $\phi=\phi'$ and fewer integrations elsewhere may reduce the execution time.

The savings in computer memory is significant for CURVENEW, since the number of matrix elements is on the order of N^2. For a given accuracy for a 0.1 λ loop, the ratio of the number of elements required for curved and linear subsections, $(N_c/N_l)^2$, is on the order of 0.1 to 0.2. This is a reduction of 80 to 90 percent. Although the memory requirements for the small loops considered here are not prohibitive for piecewise linear segments, greater memory savings will be realized for larger geometries comprised of many small features.

V. CONCLUSIONS

The use of conformal subdomain basis functions (curved subsections) to represent the current on a thin curved wire was investigated by solving the thin wire electric field integral equation using the method of moments. A solution using triangular basis functions was computer coded in FORTRAN and validated by comparing it to measured data and the results of two other method of moments solutions (LOOPSCAT and HARLOOP). The effect of varying loop radius, segment size, number of integration points and incident wave parameters on the accuracy and rate of convergence of the current expansion and backscattered field was investigated.

For small loops with circumferences on the order of a wavelength, the number of segments required to converge to a given accuracy with the curved segments was as small as 20 percent of the number of linear segments required to converge to the same accuracy (see Table 2). From computed data, it was determined that the greatest reduction in the number of unknowns for curved subsections occurs for geometries where the current amplitude variations over the surface are small and the phase variations are small or linear. As mentioned in Chapter I, the general rule of thumb for the length of one segment is 0.05 λ to 0.1 λ, which corresponds to a phase variation of 20 to 40 degrees. This restriction in phase variation is the driving factor when choosing the segment size for geometrical features having radii of curvature of approximately $\lambda/2$ or larger. A piecewise linear segment is small enough to represent the geometry accurately

in this case. The phase restriction applies to curved segments as well, but the curved segments conform exactly to the wire, and hence for small loops there is no sacrifice in geometrical accuracy by choosing segment sizes of approximately 0.05 λ or 20 electrical degrees. As the loop becomes larger, or the wavelength becomes smaller, the curved and straight subsection solutions become equivalent. The greatest advantage in using curved subsections to reduce the number of segments is for electrically small structures where small linear segments are required simply to reproduce the wire shape.

Although the number of segments was greatly reduced using conformal subsections, the execution time was increased due to the increased number of integration points per segment required for acceptable accuracy. To reduce the integration time, it is suggested that the number of integration points per wavelength be varied from a large number when evaluating the self term, to fewer points away from the self term. For certain geometries, symmetry could also be used to reduce the integration time.

To avoid singularities, the MM testing procedure was performed along the axis of the wire and the current constrained to the surface of the wire. A delta function approximation in the impedance integrations was found to reduce the time required to compute the impedance matrix, but as expected, required more segments for a given convergence accuracy.

A disadvantage in the formulation of CURVENEW is that an analytic expression for the curve is needed to perform the integrations for the impedance matrix and the excitation vector. The expressions will change each time a new curve is analyzed, and consequently, considerable effort will be required to modify the code every time a change

is made in the geometry. Programs that use linear segments are more flexible because they require only the coordinates of the points along the wire axis to generate the integration points.

The next logical step in testing the effectiveness of conformal subdomain basis functions is to formulate the solution for an equiangular spiral wire. The equiangular spiral is used in broadband antennas and has a simple mathematical form. For geometrical accuracy, the segment size in the piecewise linear formulation will be much smaller than 20 electrical degrees near the center of the spiral. Equal length conformal

TABLE 2. COMPARISON OF CURVENEW AND LOOPSCAT

	Curved Subsections		Linear Subsections	
Loop Radius	0.1λ	0.5λ	0.1λ	0.5λ
Number of Segments for 10% RMS Error, Normal Incidence	3	5	21	5
Number of Segments for 10% RMS Error, Off Axis Incidence	3	26	10	26
Execution Time*, 10% RMS Error, Normal Incidence	314 s	4669 s	32 s	2691 s
Execution Time*, 10% RMS Error, Off Axis Incidence	314 s	918 s	59 s	540 s
[Z] Matrix Size, 10 % RMS Error, Normal Incidence	9	25	441	25
[Z] Matrix Size, 10 % RMS Error, Off Axis Incidence	9	676	100	676

* Execution time measured with an IBM PC/AT.

segments may be used along the spiral arms, and it is anticipated that the number of required segments will be substantially reduced.

REFERENCES

1. Wang, J. J. H., *Generalized Moment Methods in Electromagnetics*, pp. 2-3, John Wiley & Sons, Inc., 1991.

2. Balanis, C. A., *Advanced Engineering Electromagnetics*, p 708, John Wiley & Sons, Inc., 1989.

3. Mautz, J. R. and Harrington, R. F.,"Radiation and Scattering From Bodies of Revolution," *Applied Science*, v. 20, p.405, June 1969.

4. Balanis, C. A., *Advanced Engineering Electromagnetics*, p. 689, John Wiley & Sons, Inc., 1989.

5. Balanis, C. A., *Advanced Engineering Electromagnetics*, p. 684, John Wiley & Sons, 1989.

6. Wang, J. J. H., *Generalized Moment Methods in Electromagnetics*, pp. 40-48, John Wiley & Sons, 1991.

7. Harrington, R. F., *Field Computations by Moment Methods*, Macmillan, 1968.

8. Gerald, C. F., and Wheatly, P. O., *Applied Numerical Analysis*, pp. 299-303, Addison-Wesley Publishing Company, 1989.

9. Hornbeck, Robert W., *Numerical Methods*, pp. 91-95, Quantum Publishers Inc., 1975.

10. Richmond, J. H., "A Wire Grid Model for Scattering by Conducting Bodies", *IEEE Transactions on Antennas and Propagation*, v. AP-14, no. 6, pp 782-786.

APPENDIX A

COMPUTER CODES

```
C MAIN PROGRAM  *CURVNEW.FOR*
C **** MORE NUMERICALLY EFFICIENT THAN CURVSUB.F ****
C PLANE WAVE SCATTERING FROM A CIRCULAR LOOP IN THE Z PLANE.
C METHOD OF MOMENTS WITH CURVED SUBSECTIONS
C **** RADAR CROSS SECTION CALCULATION *****
C
      COMPLEX Z(25000),B(500),C(500),BP(500),CP(500),EX,EC
      COMPLEX ET,EP,UC,RW(500),U0
      DIMENSION ECP(400),IPS(500),ANG(400),EDP(400)
      DIMENSION XG(128),AG(128),PH(500),DEL(500),PA(500)
      DIMENSION ECV(500),EXV(500),PHC(500),PHX(500)
      DATA PI/3.14159265/
      DATA IPRINT/1/,ITEST/1/
      RAD=PI/180.
      ECX=0.
      BK=2.*PI
      ETA=377.
      U0=(0.,0.)
      UC=(0.,-1.)*ETA*BK/4./PI
      CONST=16.*PI**3
      PHIRD=0.
C
C READ INPUT AND PROGRAM CONTROL PARAMETERS
C
      OPEN(1,FILE='PARAMLST.DAT')
      READ(1,*)ANGLE,DT,START,STOP,AW,R0,SEG,HARR,GCONST,XLOC,XIPOL
      LOC=INT(XLOC)
      IPOL=INT(XIPOL)
      CLOSE(1)
C
C READ GAUSSIAN CONSTANTS
C
      OPEN(2,FILE='OUTGLEG')
      READ(2,*) NG
      DO 1 I=1,NG
         READ(2,*) XG(I),AG(I)
1     CONTINUE
C
C OUTCURV IS THE FILE THAT THE SCATTERED FIELD DATA IS WRITTEN TO
C ICURVOUT IS THE FILE THAT THE LOOP CURRENT DATA IS WRITTEN TO
C
```

```
      OPEN(8,FILE='OURCURV.DAT')
      OPEN(7,FILE='ICURVOUT.DAT')
C
C GENERATE THE LOOP POINTS
C
C
C CHOOSE THE NUMBER OF POINTS BASED ON THE VALUE OF SEG
C
C     AW=AW*R0
      DPHI=SEG/R0
      NP=INT(2.*PI/DPHI)+1
      DPHI=360./FLOAT(NP)
      PH(1)=0.
      DO 10 I=2,NP+1
        PH(I)=FLOAT(I-1)*DPHI*RAD
        DEL(I-1)=(PH(I)-PH(I-1))
        PA(I-1)=(PH(I)+PH(I-1))/2.
   10 CONTINUE
      NP=NP+2
C
C OVERLAP THE ENDS SO THAT CURRENT WILL BE CONTINUOUS ON THE LOOP
C
      PH(NP)=BK+PH(2)
      DEL(NP-1)=DEL(1)
      PA(NP-1)=BK+PA(1)
      MT=NP-2
      DO 52 I=1,NP
        XHB=R0*COS(PH(I))
        YHB=R0*SIN(PH(I))
  52  CONTINUE
      WRITE(6,*) 'GEOMETRY DEFINED'
      IF(ITEST.EQ.0) GO TO 98
C
C DEFINE DIMENSIONS OF THE IMPEDANCE MATRIX BLOCKS
C
      WRITE(6,*) 'NP,MT=',NP,MT
C
C COMPUTE IMPEDANCE MATRIX ELEMENTS
C
      CALL ZCURVED(NP,R0,PH,DEL,PA,NG,XG,AG,AW,Z,ZMAX)
      DO 11 I=1,MT
         CZ=CABS(Z(I))
         AZ=ATAN2(AIMAG(Z(I)),REAL(Z(I))+1.E-8)/RAD
   11 CONTINUE
      WRITE(6,*) 'WIRE IMPEDANCE COMPUTED'
C
C PERFORM LU DECOMPOSITION
C
      CALL DECOMP(MT,IPS,Z)
      WRITE(6,*) 'Z DECOMPOSED'
```

```
C
C BEGIN FIELD CALCULATIONS
C
 98   PHR0=PHIRD*RAD
      IT=INT((STOP-START)/DT)+1
      WRITE(7,*) IT,MT,0,0
      DO 500 I=1,IT
        THETA=FLOAT(I-1)*DT+START
        THR=THETA*RAD
        PHR=PHR0
        IF(THETA.LT.180.) GO TO 99
        THR=(360.-THETA)*RAD
        PHR=PHR0+PI
 99      CONTINUE
        ET=U0
        EP=U0
C
C COMPUTE THE EXCITATION VECTOR
C
        CALL CURVEW(NP,R0,PH,DEL,PA,NG,XG,AG,THR,PHR,RW)
        IF (LOC .EQ. 0) THEN
C
C CIRCULAR POLARIZATION IF LOC=1 ELSE LINEAR
C
          IF(IPOL.EQ.1) THEN
C
C THETA POLARIZED INCIDENT WAVE (IPOL=1)
C
            DO 101 L=1,MT
 101           B(L)=RW(L)
          ELSE
C
C PHI POLARIZED INCIDENT WAVE (IPOL=2)
C
          ENDIF
        IF(ITEST.EQ.0) GO TO 9998
C
C PERFORM GAUSSIAN ELIMINATION TO DETERMINE [C]
C
        CALL SOLVE(MT,IPS,Z,B,C)
        DO 210 L=1,MT
          WRITE(7,*) L,L*DPHI,CABS(C(L)),ATAN2(REAL(C(L)),
     *    AIMAG(C(L)))/RAD
          ET=ET+RW(L)*C(L)
210      EP=EP+RW(L+MT)*C(L)
        ELSE
C
C THETA POLARIZED INCIDENT WAVE
C
          DO 221 L=1,MT
```

```
 221      B(L)=RW(L)
C
C PHI POLARIZED INCIDENT WAVE
C PHASE SHIFT FOR CP IS PI/2.
C
        DO 222 L=1,MT
222      BP(L)=RW(L+MT)*CEXP(CMPLX(0.,PI/2.))
        CALL SOLVE(MT,IPS,Z,B,C)
        CALL SOLVE(MT,IPS,Z,BP,CP)
        DO 220 L=1,MT
          WRITE(7,*) L,L*DPHI,CABS(C(L)+CP(L)),ATAN2(REAL(C(L)+CP(L)),
     *    AIMAG(C(L)+CP(L)))/RAD
          ET=ET+RW(L)*C(L)+RW(L)*CP(L)
220       EP=EP+RW(L+MT)*C(L)+RW(L+MT)*CP(L)
        ENDIF
        EC=ET*UC
        EX=EP*UC
        ANG(I)=THETA
        ECV(I)=CABS(EC)
        EXV(I)=CABS(EX)
        ECR=REAL(EC)
        ECI=AIMAG(EC)
        EXR=REAL(EX)
        EXI=AIMAG(EX)
        PHC(I)=ATAN2(ECI,ECR+1.E-20)/RAD
        PHX(I)=ATAN2(EXI,EXR+1.E-20)/RAD
        ECX=AMAX1(ECX,ECV(I),EXV(I))
  500 CONTINUE
      WRITE(6,*) 'EMAX=',ECX
      DO 600 I=1,IT
        ECV(I)=AMAX1(ECV(I),1.E-10)
        EXV(I)=AMAX1(EXV(I),1.E-10)
        ECP(I)=(ECV(I)/ECX)**2
        EDP(I)=(EXV(I)/ECX)**2
        ECP(I)=AMAX1(ECP(I),.00001)
        EDP(I)=AMAX1(EDP(I),.00001)
        ECP(I)=10.*ALOG10(ECP(I))
        EDP(I)=10.*ALOG10(EDP(I))
  600 CONTINUE
      SIGMA=(ECX**2)*CABS(UC)/(2.*BK)
      SIGDB=10.*ALOG10(SIGMA)
      WRITE(6,*) 'BACKSCATTER CROSS-SECTION, IN DB=',SIGMA,SIGDB
208   FORMAT(/,5X,'SIGMA/WAVL SQ=',E15.4,
     * /,5X,'        IN DB=',F8.4)
      DO 9000 L=1,IT
        WRITE(8,*) ANG(L),ECV(L)
9000  CONTINUE
9998  STOP
      END
      SUBROUTINE SOLVE(N,IPS,UL,B,X)
```

```
C
C
C SUBROUTINE TO SOLVE SYSTEM OF EQUATIONS WITH COMPLEX
COEFFICIENTS.
C CALL 'DECOMP' FIRST. (FROM MAUTZ AND HARRINGTON)
C
      COMPLEX UL(50000),B(200),X(200),SUM
      DIMENSION IPS(500)
      NP1=N+1
      IP=IPS(1)
      X(1)=B(IP)
      DO 2 I=2,N
      IP=IPS(I)
      IPB=IP
      IM1=I-1
      SUM=(0.,0.)
      DO 1 J=1,IM1
      SUM=SUM+UL(IP)*X(J)
    1 IP=IP+N
    2 X(I)=B(IPB)-SUM
      K2=N*(N-1)
      IP=IPS(N)+K2
      X(N)=X(N)/UL(IP)
      DO 4 IBACK=2,N
      I=NP1-IBACK
      K2=K2-N
      IPI=IPS(I)+K2
      IP1=I+1
      SUM=(0.,0.)
      IP=IPI
      DO 3 J=IP1,N
      IP=IP+N
    3 SUM=SUM+UL(IP)*X(J)
    4 X(I)=(X(I)-SUM)/UL(IPI)
      RETURN
      END
      SUBROUTINE DECOMP(N,IPS,UL)
C
C SUBROUTINE TO DECOMPOSE SYSTEM OF EQUATIONS.
C FROM MAUTZ AND HARRINGTON.
C
      COMPLEX UL(50000),PIVOT,EM
      DIMENSION SCL(200),IPS(200)
      DO 5 I=1,N
      IPS(I)=I
      RN=0.
      J1=I
      DO 2 J=1,N
      ULM=ABS(REAL(UL(J1)))+ABS(AIMAG(UL(J1)))
      J1=J1+N
```

```
      IF(RN-ULM) 1,2,2
    1 RN=ULM
    2 CONTINUE
      SCL(I)=1./RN
    5 CONTINUE
      NM1=N-1
      K2=0
      DO 17 K=1,NM1
      BIG=0.
      DO 11 I=K,N
      IP=IPS(I)
      IPK=IP+K2
      SIZE=(ABS(REAL(UL(IPK)))+ABS(AIMAG(UL(IPK))))*SCL(IP)
      IF(SIZE-BIG) 11,11,10
   10 BIG=SIZE
      IPV=I
   11 CONTINUE
      IF(IPV-K) 14,15,14
   14 J=IPS(K)
      IPS(K)=IPS(IPV)
      IPS(IPV)=J
   15 KPP=IPS(K)+K2
      PIVOT=UL(KPP)
      KP1=K+1
      DO 16 I=KP1,N
      KP=KPP
      IP=IPS(I)+K2
      EM=-UL(IP)/PIVOT
   18 UL(IP)=-EM
      DO 16 J=KP1,N
      IP=IP+N
      KP=KP+N
      UL(IP)=UL(IP)+EM*UL(KP)
   16 CONTINUE
      K2=K2+N
   17 CONTINUE
      RETURN
      END
      SUBROUTINE ZCURVED(NP,R0,PH,DEL,PA,NG,XG,AG,A,Z,ZMAX)
C
C IMPEDANCE ELEMENTS FOR CURVED BASIS FUNCTIONS.
C SPECIFICALLY DERIVED FOR A CIRCULAR LOOP -- NOT A GENERAL CASE.
C
      COMPLEX CEXP,Z(50000),CON,CMPLX,SUMA,SUMB
      COMPLEX U0,SUM1,SUM2,SUM3,SUM4,EXP,ZT(500)
      DIMENSION DEL(500),PH(500),XG(128),AG(128),PA(500)
C      OPEN(2,FILE='ZCURV.DAT')
      ETA=377.
      ZMAX=0.
      PI=3.14159
```

```
      BK=2.*PI
      BK2=BK**2
      U0=(0.,0.)
      CON=(0.,1.)*BK*ETA/(4.*PI)*R0**2
      NT=NP-2
C
C COMPUTE Z(1,LQ) = ZT(LQ)
C
      KQ=1
      P1=DEL(KQ)/2.
      P2=PA(KQ)
      P3=DEL(KQ+1)/2.
      P4=PA(KQ+1)
C
C DO THE L LOOP
C
      DO 600 LQ=1,NT
        PP1=DEL(LQ)/2.
        PP2=PA(LQ)
        PP3=DEL(LQ+1)/2.
        PP4=PA(LQ+1)
C
C DO THE PHI INTEGRATION
C *** FIRST PART FROM PHI(K) TO PHI(K+1)
C PHI PRIMED INTEGRATION FOR THE POSITIVE SLOPE OF LQ
C
        SUMA=U0
        PHA=PA(KQ)
        DO 100 I=1,NG
          PHI=P1*XG(I)+P2
          TK=(PHI-PH(KQ))/DEL(KQ)
          TKP=1./DEL(KQ)/R0
          SUM1=U0
          DO 90 J=1,NG
            PHIP=PP1*XG(J)+PP2
            TL=(PHIP-PH(LQ))/DEL(LQ)
            TLP=1./DEL(LQ)/R0
            CC=COS(PHI-PHIP)
C
C COMPUTE THE MAGNITUDE OF R. NOTE THAT R IS COMPUTED FROM THE
C WIRE AXIS TO THE SURFACE OF THE WIRE
C
            RR=R0*SQRT(4.*(SIN((PHI-PHIP)/2.))**2+(A/R0)**2)
            EXP=CEXP(CMPLX(0.,-BK*RR))/RR
            SUM1=SUM1+AG(J)*EXP*(TK*TL*CC-TKP*TLP/BK2)
90         CONTINUE
          SUM1=SUM1*PP1
C
C PHI PRIMED INTEGRATION FOR THE NEGATIVE SLOPE OF LQ
C
```

```
          SUM2=U0
          DO 80 J=1,NG
            PHIP=PP3*XG(J)+PP4
            TL=1.-(PHIP-PH(LQ+1))/DEL(LQ+1)
            TLP=-1./DEL(LQ+1)/R0
            CC=COS(PHI-PHIP)
C
C COMPUTE THE MAGNITUDE OF R. NOTE THAT R IS COMPUTED FROM THE
C WIRE AXIS TO THE SURFACE OF THE WIRE
C
            RR=R0*SQRT(4.*(SIN((PHI-PHIP)/2.))**2+(A/R0)**2)
            EXP=CEXP(CMPLX(0.,-BK*RR))/RR
            SUM2=SUM2+AG(J)*EXP*(TK*TL*CC-TKP*TLP/BK2)
80          CONTINUE
          SUM2=SUM2*PP3
          SUMA=SUMA+(SUM1+SUM2)*AG(I)
100     CONTINUE
        SUMA=SUMA*P1
C
C *** SECOND PART FROM PHI(K+1) TO PHI(K+2)
C PHI PRIMED INTEGRATION FOR THE POSITIVE SLOPE OF LQ
C
        SUMB=U0
        PHA=PA(KQ+1)
        DO 101 I=1,NG
          PHI=P3*XG(I)+P4
          TK=1.-(PHI-PH(KQ+1))/DEL(KQ+1)
          TKP=-1./DEL(KQ+1)/R0
          SUM3=U0
          DO 91 J=1,NG
            PHIP=PP1*XG(J)+PP2
            TL=(PHIP-PH(LQ))/DEL(LQ)
            TLP=1./DEL(LQ)/R0
            CC=COS(PHI-PHIP)
C
C COMPUTE THE MAGNITUDE OF R. NOTE THAT R IS COMPUTED FROM THE
C WIRE AXIS TO THE SURFACE OF THE WIRE
C
            RR=R0*SQRT(4.*(SIN((PHI-PHIP)/2.))**2+(A/R0)**2)
            EXP=CEXP(CMPLX(0.,-BK*RR))/RR
            SUM3=SUM3+AG(J)*EXP*(TK*TL*CC-TKP*TLP/BK2)
91          CONTINUE
          SUM3=SUM3*PP1
C
C PHI PRIMED INTEGRATION FOR THE NEGATIVE SLOPE OF LQ
C
          SUM4=U0
          DO 81 J=1,NG
            PHIP=PP3*XG(J)+PP4
            TL=1.-(PHIP-PH(LQ+1))/DEL(LQ+1)
```

```
              TLP=-1./DEL(LQ+1)/R0
              CC=COS(PHI-PHIP)
C
C COMPUTE THE MAGNITUDE OF R. NOTE THAT R IS COMPUTED FROM THE
C WIRE AXIS TO THE SURFACE OF THE WIRE
C
              RR=R0*SQRT(4.*(SIN((PHI-PHIP)/2.))**2+(A/R0)**2)
              EXP=CEXP(CMPLX(0.,-BK*RR))/RR
              SUM4=SUM4+AG(J)*EXP*(TK*TL*CC-TKP*TLP/BK2)
81         CONTINUE
            SUM4=SUM4*PP3
            SUMB=SUMB+(SUM3+SUM4)*AG(I)
101       CONTINUE
          SUMB=SUMB*P3
          ZT(LQ)=CON*(SUMA+SUMB)
          ZMAX=AMAX1(ZMAX,CABS(ZT(LQ)))
600     CONTINUE
        ZT(NT)=ZT(2)
C
C FILL THE ENTIRE Z MATRIX USING SYMMETRY PROPERTIES
C [Z] IS A SYMMETRICAL TOEPLITZ MATRIX
C ROW INDEX, I; COL INDEX, J
C
        DO 10 I=1,NT
          DO 10 J=1,NT
            K=(I-1)*NT+J
            Z(K)=U0
            IJ=IABS(I-J)
            IF(IJ.GT.NT) GO TO 10
            IJ1=IJ+1
            Z(K)=ZT(IJ1)
10      CONTINUE
        RETURN
        END
        SUBROUTINE CURVEW(NP,R0,PH,DEL,PA,NG,XG,AG,THR,PHR,R)
C
C  PLANE WAVE EXCITATION VECTOR ELEMENTS FOR A LOOP USING CURVED
C  BASIS FUNCTIONS. INCIDENCE DIRECTION IS (THR,PHR).  THE WIRE LIES
C  IN THE X-Y PLANE (Z=0).
C
        COMPLEX U0,R(500),SUM1,SUM2,SUM3,SUM4,CEXP,FF
        COMPLEX GG,CMPLX,SUMT,SUMP
        DIMENSION PH(500),DEL(500),PA(500),XG(128),AG(128)
        MT=NP-2
        U0=(0.,0.)
        PI=3.14159
        BK=2.*PI
        ST=SIN(THR)
        CT=COS(THR)
        DO 50 IP=1,MT
```

```
          SUM1=U0
          SUM2=U0
          SUM3=U0
          SUM4=U0
          P1=DEL(IP)/2.
          P2=PA(IP)
          P3=DEL(IP+1)/2.
          P4=PA(IP+1)
          DO 20 I=1,NG
            PHI=P1*XG(I)+P2
            CC=COS(PHR-PHI)
            SS=SIN(PHR-PHI)*CT
            FF=AG(I)*(PHI-PH(IP))/DEL(IP)*CEXP(CMPLX(0.,BK*R0*ST*CC))
            SUM1=SUM1+CC*FF
            SUM2=SUM2+SS*FF
            PHI=P3*XG(I)+P4
            CC=COS(PHR-PHI)
            SS=SIN(PHR-PHI)*CT
            GG=AG(I)*(1.-(PHI-PH(IP+1))/DEL(IP+1))*CEXP(CMPLX(0.,
     *      BK*R0*ST*CC))
            SUM3=SUM3+CC*GG
            SUM4=SUM4+SS*GG
 20       CONTINUE
          SUMP=SUM1*P1+SUM3*P3
          SUMT=SUM2*P1+SUM4*P3
C
C R-WIRE-THETA IN R(IP) AND R-WIRE-PHI IN R(IP+MT)
C
          R(IP)=SUMT*R0
          R(IP+MT)=SUMP*R0
 50     CONTINUE
        RETURN
        END
```

```
C MAIN PROGRAM  *LOOP.FOR*
C PLANE WAVE SCATTERING FROM A CIRCULAR LOOP IN THE Z PLANE.
C **** RADAR CROSS SECTION CALCULATION *****
C
      COMPLEX Z(15000),B(500),C(500),BP(500),CP(500),U
      COMPLEX ET,EP,UC,RW(500),U0
      DIMENSION ECP(400),IPS(500),ANG(400),EDP(400)
      DIMENSION ZH(200),XT(128),AT(128),XH(200),YH(200)
      DIMENSION ECV(400),EXV(400),PHC(400),PHX(400)
      DATA PI/3.14159265/
      DATA IPRINT/1/
C
C READ INPUT AND PROGRAM CONTROL PARAMETERS
C
      OPEN(1,FILE='PARAMLST.DAT')
      READ(1,*)ANGLE,DT,START,STOP,AW,RB,SEG,HARR,GCONST,XLOC,XIPOL
      LOC=INT(XLOC)
      IPOL=INT(XIPOL)
      CLOSE(1)
C
C READ GAUSSIAN CONSTANTS
C
      OPEN(2,FILE='OUTGLEG')
      READ(2,*) NT
      DO 2 I=1,NT
        READ(2,*) XT(I),AT(I)
 2    CONTINUE
      RAD=PI/180.
      ECX=0.
      BK=2.*PI
      ETA=377.
      U=(0.,1.)
      U0=(0.,0.)
      UC=-U*ETA*BK/4./PI
      CONST=16.*PI**3
      NT2=NT/2
C
C OUTLOOP IS THE FILE THAT THE SCATTERED FIELD DATA IS WRITTEN TO
C ISTOUT IS THE FILE THAT THE LOOP CURRENT DATA IS WRITTEN TO
C
      OPEN(8,FILE='OUTLOOP.DAT')
      OPEN(7,FILE='ISTOUT.DAT')
      DPHI=SEG/RB
      NP=INT(2.*PI/DPHI)+1
      DPHI=360./FLOAT(NP)
      WRITE(6,*) 'DPHI,NP=',DPHI,NP
C
C GENERATE THE LOOP POINTS. MULTIPLY ALL QUANTITIES BY BK (=2*PI)
C
C
```

```
C CHOOSE THE NUMBER OF POINTS BASED ON THE VALUE OF SEG
C
      AK=AW*BK
      DO 10 I=1,NP+1
      PP=FLOAT(I-1)*DPHI*RAD
      XH(I)=RB*COS(PP)*BK
      YH(I)=RB*SIN(PP)*BK
      ZH(I)=0.
10    CONTINUE
      NP=NP+2
C
C OVERLAP THE ENDS SO THAT THE CURRENT IS CONTINUOUS
C
      XH(NP)=XH(2)
      YH(NP)=YH(2)
      ZH(NP)=ZH(2)
      DO 52 I=1,NP
        YHB=YH(I)/BK
        XHB=XH(I)/BK
        DEL=0.
        IF(I.NE.1) THEN
          DXX=XHB-XH(I-1)/BK
          DYY=YHB-YH(I-1)/BK
          DEL=SQRT(DXX**2+DYY**2)
        ENDIF
52    CONTINUE
C
C DEFINE DIMENSIONS OF THE IMPEDANCE MATRIX BLOCKS
C
      MT=NP-2
      WRITE(6,*) 'MT=',MT
C
C COMPUTE IMPEDANCE MATRIX ELEMENTS
C
      CALL ZMATWW(1,1,NP,RB,XH,YH,ZH,NT,XT,AT,AK,Z)
      WRITE(6,*) 'Z COMPUTED'
      IF(IPRINT.EQ.0) THEN
        DO 1010 I=1,MT
          CZ=CABS(Z(I))
          AZ=ATAN2(AIMAG(Z(I)),REAL(Z(I))+1.E-8)/RAD
          WRITE(6,*) 'I,Z=',I,Z(I),CZ,CZ/ZMAX,AZ
1010      CONTINUE
      ENDIF
C
C PERFORM LU DECOMPOSITION
C
      CALL DECOMP(MT,IPS,Z)
      WRITE(6,*) 'Z DECOMPOSED'
C
C BEGIN FIELD CALCULATIONS.  PHI FOR PATTERN CUT (DEGREES)=PHID
```

```
C
      PHID=0.
      PHR0=PHID*RAD
      IT=INT((STOP-START)/DT)+1
      WRITE(7,*) IT,MT,0,0
      DO 500 I=1,IT
        THETA=FLOAT(I-1)*DT+START
        THR=THETA*RAD
        PHR=PHR0
        IF(THETA.LT.180.) GO TO 99
        THR=(360.-THETA)*RAD
        PHR=PHR0+PI
   99   CONTINUE
        ET=U0
        EP=U0
C
C COMPUTE THE EXCITATION VECTOR
C
        CALL PLANEW(NP,XH,YH,ZH,THR,PHR,RW)
        IF (LOC .EQ. 0) THEN
C
C CIRCULAR POLARIZATION IF LOC=1 ELSE LINEAR
C
          IF(IPOL.EQ.1) THEN
C
C THETA POLARIZED INCIDENT WAVE (IPOL=1)
C
            DO 101 L=1,MT
101           B(L)=RW(L)
        ELSE
C
C PHI POLARIZED INCIDENT WAVE (IPOL=2)
C
            DO 102 L=1,MT
102           B(L)=RW(L+MT)
        ENDIF
C
C PERFORM GAUSSIAN ELIMINATION TO DETERMINE [C]
C
        CALL SOLVE(MT,IPS,Z,B,C)
        DO 210 L=1,MT
          WRITE(7,*) L,L*DPHI,CABS(C(L)),ATAN2(REAL(C(L)),
     *    AIMAG(C(L)))/RAD
          ET=ET+(RW(L)/BK)*C(L)
210     EP=EP+(RW(L+MT)/BK)*C(L)
        ELSE
C
C THETA PLOARIZED INCIDENT WAVE
C
          DO 221 L=1,MT
```

```
 221     B(L)=RW(L)
C
C PHI POLARIZED INCIDENT WAVE
C PHASE SHIFT FOR CP IS PI/2.
C
        DO 222 L=1,MT
 222     BP(L)=RW(L+MT)*CEXP(CMPLX(0.,PI/2.))
        CALL SOLVE(MT,IPS,Z,B,C)
        CALL SOLVE(MT,IPS,Z,BP,CP)
        DO 220 L=1,MT
          WRITE(7,*) L,L*DPHI,CABS(C(L)+CP(L)),ATAN2(REAL(C(L)+CP(L)),
     *    AIMAG(C(L)+CP(L)))/RAD
          ET=ET+RW(L)*C(L)+RW(L)*CP(L)
220       EP=EP+RW(L+MT)*C(L)+RW(L+MT)*CP(L)
      ENDIF
      ET=UC*ET
      EP=UC*EP
C
C E-THETA IS CO-POL; E-PHI IS CROSS-POL
C
      ANG(I)=THETA
      ECV(I)=CABS(ET)
      EXV(I)=CABS(EP)
      ECR=REAL(ET)
      ECI=AIMAG(ET)
      EXR=REAL(EP)
      EXI=AIMAG(EP)
      PHC(I)=ATAN2(ECI,ECR+1.E-20)/RAD
      PHX(I)=ATAN2(EXI,EXR+1.E-20)/RAD
      ECX=AMAX1(ECX,ECV(I),EXV(I))
500  CONTINUE
     DO 600 I=1,IT
      ECV(I)=AMAX1(ECV(I),1.E-10)
      EXV(I)=AMAX1(EXV(I),1.E-10)
      ECP(I)=(ECV(I)/ECX)**2
      EDP(I)=(EXV(I)/ECX)**2
      ECP(I)=AMAX1(ECP(I),.00001)
      EDP(I)=AMAX1(EDP(I),.00001)
      ECP(I)=10.*ALOG10(ECP(I))
      EDP(I)=10.*ALOG10(EDP(I))
600  CONTINUE
     SIGMA=(ECX**2)*CABS(UC)/(2.*BK)
     SIGDB=10.*ALOG10(SIGMA)
     WRITE(6,*) 'SIGMA, IN DB=',SIGMA,SIGDB
     DO 9000 L=1,IT
      WRITE(8,*) ANG(L),ECV(L)
9000 CONTINUE
900  CONTINUE
     STOP
     END
```

```
      SUBROUTINE SOLVE(N,IPS,UL,B,X)
C
C SUBROUTINE TO SOLVE SYSTEM OF EQUATIONS WITH COMPLEX
COEFFICIENTS.
C CALL 'DECOMP' FIRST. (FROM MAUTZ AND HARRINGTON)
C
      COMPLEX UL(50000),B(500),X(500),SUM
      DIMENSION IPS(500)
      NP1=N+1
      IP=IPS(1)
      X(1)=B(IP)
      DO 2 I=2,N
      IP=IPS(I)
      IPB=IP
      IM1=I-1
      SUM=(0.,0.)
      DO 1 J=1,IM1
      SUM=SUM+UL(IP)*X(J)
    1 IP=IP+N
    2 X(I)=B(IPB)-SUM
      K2=N*(N-1)
      IP=IPS(N)+K2
      X(N)=X(N)/UL(IP)
      DO 4 IBACK=2,N
      I=NP1-IBACK
      K2=K2-N
      IPI=IPS(I)+K2
      IP1=I+1
      SUM=(0.,0.)
      IP=IPI
      DO 3 J=IP1,N
      IP=IP+N
    3 SUM=SUM+UL(IP)*X(J)
    4 X(I)=(X(I)-SUM)/UL(IPI)
      RETURN
      END
      SUBROUTINE DECOMP(N,IPS,UL)
C
C SUBROUTINE TO DECOMPOSE SYSTEM OF EQUATIONS.
C FROM MAUTZ AND HARRINGTON.
C
      COMPLEX UL(50000),PIVOT,EM
      DIMENSION SCL(500),IPS(500)
      DO 5 I=1,N
      IPS(I)=I
      RN=0.
      J1=I
      DO 2 J=1,N
      ULM=ABS(REAL(UL(J1)))+ABS(AIMAG(UL(J1)))
      J1=J1+N
```

```
      IF(RN-ULM) 1,2,2
    1 RN=ULM
    2 CONTINUE
      SCL(I)=1./RN
    5 CONTINUE
      NM1=N-1
      K2=0
      DO 17 K=1,NM1
      BIG=0.
      DO 11 I=K,N
      IP=IPS(I)
      IPK=IP+K2
      SIZE=(ABS(REAL(UL(IPK)))+ABS(AIMAG(UL(IPK))))*SCL(IP)
      IF(SIZE-BIG) 11,11,10
   10 BIG=SIZE
      IPV=I
   11 CONTINUE
      IF(IPV-K) 14,15,14
   14 J=IPS(K)
      IPS(K)=IPS(IPV)
      IPS(IPV)=J
   15 KPP=IPS(K)+K2
      PIVOT=UL(KPP)
      KP1=K+1
      DO 16 I=KP1,N
      KP=KPP
      IP=IPS(I)+K2
      EM=-UL(IP)/PIVOT
   18 UL(IP)=-EM
      DO 16 J=KP1,N
      IP=IP+N
      KP=KP+N
      UL(IP)=UL(IP)+EM*UL(KP)
   16 CONTINUE
      K2=K2+N
   17 CONTINUE
      RETURN
      END
      SUBROUTINEZMATWW(NWIRES,NW1,NW2,RB,XH,YH,ZH,NT,XT,AT,AK,ZZ)
C
C IMPEDANCE ELEMENTS FOR LINEAR BASIS FUNCTIONS.
C
      COMPLEX CEXP,Z(200),ZZ(15000),CON,CMPLX,EXP
      COMPLEX U0,SUM,SUM1,SUM2,SUM3,SUM4
      DIMENSION ZH(200),XT(128),AT(128).XH(200),YH(200),UU(200)
      DIMENSION D1(200),S1(200),C1(200),ZS1(200)
      DIMENSION XS1(200),YS1(200)
      DIMENSION CU(200),SU(200)
      INTEGER NT,NWIRES,NW1(4),NW2(4),NS(4)
      PI=3.1415926
```

```
      PI2=2.*PI
      ETA=377.
      BK=PI2
      U0=(0.,0.)
      CON=(0.,1.)*BK*ETA/(4.*PI)
      A=AK
C
C DEFINE GEOMETRY TERMS FOR THE WIRE. XH,YH,ZH ARE ALL KNOWNS.
C
      DO 5 L=1,NWIRES
 5    NS(L)=NW2(L)-NW1(L)+1
      NS1=NW2(NWIRES)-NW1(1)
      NPS=NS1+1
      NTRIA=NPS-2
      DO 10 N=2,NPS
        N0=N-1
        I=NW1(1)+N-1
        I2=I-1
C
C AVERAGE VALUES
C
        ZS1(N0)=.5*(ZH(I)+ZH(I2))
        XS1(N0)=.5*(XH(I)+XH(I2))
        YS1(N0)=.5*(YH(I)+YH(I2))
        DX=XH(I)-XH(I2)
        DY=YH(I)-YH(I2)
        D1(N0)=SQRT(DX**2+DY**2)
        UU(N0)=ATAN2(DY,DX+1.E-5)
        CU(N0)=COS(UU(N0))
        SU(N0)=SIN(UU(N0))
        S1(N0)=DR/D1(N0)
        C1(N0)=DZ/D1(N0)
   10 CONTINUE
      IP=1
      WRITE(6,*) 'IP=',IP
      DO 600 JQ=1,NTRIA
C
C DOING I1
C
        I=IP
        J=JQ
        CC=COS(UU(I)-UU(J))
        TIP=1./D1(I)
        TJP=1./D1(J)
        T1=D1(I)/2.
        T2=D1(J)/2.
        SUM=U0
        DO 100 K=1,NT
          T=T1*XT(K)
          TI=.5+T/D1(I)
```

```
          XI=XS1(I)+T*CU(I)
          YI=YS1(I)+T*SU(I)
          ZI=ZS1(I)
          DO 100 L=1,NT
            TP=T2*XT(L)
            TJ=.5+TP/D1(J)
            XJ=XS1(J)+TP*CU(J)
            YJ=YS1(J)+TP*SU(J)
            ZJ=ZS1(J)
            RP=SQRT((XI-XJ)**2+(YI-YJ)**2+A**2)
            EXP=CEXP(CMPLX(0.,-RP))/RP
            SUM=SUM+AT(L)*AT(K)*EXP*(TI*TJ*CC-TIP*TJP)
100     CONTINUE
        SUM1=SUM*T1*CON*T2
C
C DOING I2
C
        J=JQ+1
        CC=COS(UU(I)-UU(J))
        TIP=1./D1(I)
        TJP=-1./D1(J)
        T1=D1(I)/2.
        T2=D1(J)/2.
        SUM=U0
        DO 101 K=1,NT
          T=T1*XT(K)
          TI=.5+T/D1(I)
          XI=XS1(I)+T*CU(I)
          YI=YS1(I)+T*SU(I)
          ZI=ZS1(I)
          DO 101 L=1,NT
            TP=T2*XT(L)
            TJ=.5-TP/D1(J)
            XJ=XS1(J)+TP*CU(J)
            YJ=YS1(J)+TP*SU(J)
            ZJ=ZS1(J)
            RP=SQRT((XI-XJ)**2+(YI-YJ)**2+A**2)
            EXP=CEXP(CMPLX(0.,-RP))/RP
            SUM=SUM+AT(L)*AT(K)*EXP*(TI*TJ*CC-TIP*TJP)
101     CONTINUE
        SUM2=SUM*T1*CON*T2
C
C DOING I3
C
        I=IP+1
        J=JQ
        CC=COS(UU(I)-UU(J))
        TIP=-1./D1(I)
        TJP=1./D1(J)
        T1=D1(I)/2.
```

```
      T2=D1(J)/2.
      SUM=U0
      DO 102 K=1,NT
        T=T1*XT(K)
        TI=.5-T/D1(I)
        XI=XS1(I)+T*CU(I)
        YI=YS1(I)+T*SU(I)
        ZI=ZS1(I)
        DO 102 L=1,NT
          TP=T2*XT(L)
          TJ=.5+TP/D1(J)
          XJ=XS1(J)+TP*CU(J)
          YJ=YS1(J)+TP*SU(J)
          ZJ=ZS1(J)
          RP=SQRT((XI-XJ)**2+(YI-YJ)**2+A**2)
          EXP=CEXP(CMPLX(0.,-RP))/RP
          SUM=SUM+AT(L)*AT(K)*EXP*(TI*TJ*CC-TIP*TJP)
102     CONTINUE
      SUM3=SUM*T1*CON*T2
C
C DOING I4
C
      J=JQ+1
      CC=COS(UU(I)-UU(J))
      TIP=-1./D1(I)
      TJP=-1./D1(J)
      T1=D1(I)/2.
      T2=D1(J)/2.
      SUM=U0
      DO 103 K=1,NT
        T=T1*XT(K)
        TI=.5-T/D1(I)
        XI=XS1(I)+T*CU(I)
        YI=YS1(I)+T*SU(I)
        ZI=ZS1(I)
        DO 103 L=1,NT
          TP=T2*XT(L)
          TJ=.5-TP/D1(J)
          XJ=XS1(J)+TP*CU(J)
          YJ=YS1(J)+TP*SU(J)
          ZJ=ZS1(J)
          RP=SQRT((XI-XJ)**2+(YI-YJ)**2+A**2)
          EXP=CEXP(CMPLX(0.,-RP))/RP
          SUM=SUM+AT(L)*AT(K)*EXP*(TI*TJ*CC-TIP*TJP)
103     CONTINUE
      SUM4=SUM*T1*CON*T2
C
C IMPEDANCE ELEMENT FOR IP,JQ
C
      KK=(JQ-1)*NTRIA+IP
```

```
        Z(JQ)=(SUM1+SUM2+SUM3+SUM4)
600   CONTINUE
      Z(NTRIA)=Z(2)
C
C FILL THE ENTIRE Z MATRIX USING SYMMETRY PROPERTIES
C ROW INDEX, I; COL INDEX, J
C
      DO 12 I=1,NTRIA
        DO 12 J=1,NTRIA
          K=(I-1)*NTRIA+J
          ZZ(K)=U0
          IJ=IABS(I-J)
          IF(IJ.GT.NTRIA) GO TO 12
          IJ1=IJ+1
          ZZ(K)=Z(IJ1)
12    CONTINUE
      CLOSE(2)
      RETURN
      END
      SUBROUTINE PLANEW(NP,XH,YH,ZH,THR,PHR,R)
C
C  PLANE WAVE EXCITATION VECTOR ELEMENTS FOR WIRE AND
C  INCIDENCE DIRECTION IS (THR.PHR).
C  WIRE LIES IN THE X-Y PLANE (Z=0)
C
      COMPLEX U0,C,R(2000),CEXP,EXP,FI1,FI2,SI,DI,CMPLX
      DIMENSION ZH(500),XH(500),YH(500)
      MP2=NP-1
      MT2=NP-2
      U0=(0.,0.)
      CC=COS(THR)
      SS=SIN(THR)
      CP=COS(PHR)
      SP=SIN(PHR)
      UP=SS*CP
      VP=SS*SP
      DO 12 IP=1,MP2
        II=IP
        I=II+1
        ZS=.5*(ZH(I)+ZH(II))
        XS=.5*(XH(I)+XH(II))
        YS=.5*(YH(I)+YH(II))
        DX=XH(I)-XH(II)
        DY=YH(I)-YH(II)
        D1=SQRT(DX**2+DY**2)
        SU=DY/D1
        CU=DX/D1
C FOR WIRES IN THE XY PLANE SIN(V)=1 AND COS(V)=0
        SV=1.0
        CV=0.0
```

```
C
C WIRE SEGMENT CALCULATIONS
C
      A=UP*CU+VP*SU
      B=UP*XS+VP*YS
      C=CMPLX(0.,A)
      EXP=CEXP(CMPLX(0.,B))
      AA=CC*(CU*CP+SU*SP)
      BB=SU*CP-SP*CU
      PSI=D1*A/2.
      IF(PSI.NE.0.) GO TO 60
      SINC=1.
      GO TO 61
   60 SINC=SIN(PSI)/PSI
   61 COSP=COS(PSI)
      FI1=SINC*D1*EXP/2.
      FI2=(0.,0.)
      IF(ABS(A).LT.1.E-4) GO TO 62
      CSP=COSP-SINC
      IF(ABS(CSP).LT.1.E-4) GO TO 62
      FI2=EXP/C*CSP
   62 CONTINUE
      SI=FI1+FI2
      DI=FI1-FI2
C
C R-WIRE-THETA
C
      IF(IP.EQ.MP2) GO TO 10
      R(IP)=AA*SI
      R(IP+MT2)=BB*SI
   10 CONTINUE
C
C R-WIRE-PHI
C
   14 IF(IP.EQ.1) GO TO 12
      R(IP-1)=R(IP-1)+AA*DI
      R(IP-1+MT2)=R(IP-1+MT2)+BB*DI
   12 CONTINUE
  210 RETURN
      END
```

```
C MAIN PROGRAM  *HARLOOP.F*
C PLANE WAVE SCATTERING FROM A CIRCULAR LOOP IN THE Z PLANE.
C **** RADAR CROSS SECTION CALCULATION *****
C USING HARRINGTON'S FORMULATION FROM THE BOOK 'FIELD COMP. BY
C MM' (P.83 TO 95)
C
      COMPLEX Z(5000),E(250),C(250),EX,EC,ET,EP,RW(1000),UC,EPHI(500)
      COMPLEX CPHI(500)
      DIMENSION ECP(500),IPS(250),ANG(500),EDP(500),XT(300),AT(300)
      DIMENSION ECV(500),EXV(500),PHC(500),PHX(500)
      DATA PI/3.14159265/
      DATA IPRINT/1/,ITEST/1/
      RAD=PI/180.
      ECX=0.
      BK=2.*PI
      ETA=377.
      UC=(0,-1)*ETA*BK/(4.*PI)
      PHIRD=0.
      OPEN(1,FILE='PARAMLST.DAT')
      READ(1,*) ANGLE,DT,START,STOP,A,B,SEG,AHARR,GCONST,XLOC,XIPOL
      IPOL=INT(XIPOL)
      LOC=INT(XLOC)
      CLOSE(1)
      NM=INT(AHARR)
      CON=(377.*BK)**2/2./BK
      OPEN(2,FILE='OUTGLEG')
      READ(2,*) NT
      DO 1 I=1,NT
        READ(2,*) XT(I),AT(I)
1     CONTINUE
      OPEN(8,FILE='OUTHARR.DAT')
      OPEN(7,FILE='IHARROUT.DAT')
      NROW=2*NM+1
      WRITE(6,1300) B,A,NROW,NT
 1300 FORMAT(//,5X,'PLANE WAVE SCATTERING BY A CIRCULAR LOOP',/
     * ,5X,        'USING METHOD IN HARRINGTONS MM TEXT BOOK',/
     * ,5X,'LOOP RADIUS (WAVL)=',F8.4,/,5X,'WIRE RADIUS (WAVL)=',
     * F8.4,/,5X,'NUMBER OF AZIMUTHAL MODES (INCLUDING ZERO)=',I3,
     * /,5X,'NUMBER OF INTEGRATION POINTS IN PHI=',I4)
C
C COMPUTE IMPEDANCE MATRIX ELEMENTS
C
      WRITE(6,*) 'CALLING ZMAT'
      CALL ZMATWW(NM,A,B,NT,XT,AT,Z)
      WRITE(6,*) 'WIRE IMPEDANCE COMPUTED'
      CALL DECOMP(NROW,IPS,Z)
      WRITE(6,*) 'Z DECOMPOSED'
C
C BEGIN FIELD CALCULATIONS
C
```

```
      PHR0=PHIRD*RAD
      IT=INT((STOP-START)/DT)+1
      WRITE(7,*) IT,NROW
      DO 500 I=1,IT
        THETA=FLOAT(I-1)*DT+START
        WRITE(6,*) 'THETA=',THETA
        THR=THETA*RAD
        PHR=PHR0
        IF(THETA.LT.180.) GO TO 99
        THR=(360.-THETA)*RAD
        PHR=PHR0+PI
   99   CONTINUE
        ET=(0.,0.)
        EP=(0.,0.)
        CALL PLANEW(NM,B,THR,PHR,RW)
C TRANSMIT VECTOR ELEMENTS ARE TRANSPOSED FORMS OF RECEIVE
VECTOR
C ALSO THE THETA COMPONENT GETS A NEGATIVE SIGN
        IF(LOC .EQ. 0) THEN
          IF(IPOL.EQ.1) THEN
C
C THETA POLARIZED INCIDENT WAVE (IPOL=1)
C
            DO 101 L=1,NROW
101           E(NROW-L+1)=RW(L)
            ELSE
C
C PHI POLARIZED INCIDENT WAVE (IPOL=2)
C
            DO 102 L=1,NROW
102           E(NROW-L+1)=RW(L+NROW)
          ENDIF
          WRITE(6,*) 'CALLING SOLVE'
          CALL SOLVE(NROW,IPS,Z,E,C)
          WRITE(6,*) 'RETURNED FROM SOLVE'
          DO 210 L=1,NROW
            WRITE(7,*) C(L)
            ET=ET+RW(L)*C(L)
  210       EP=EP+RW(L+NROW)*C(L)
        ELSE
C
C E-THETA IS CO-POL; E-PHI IS CROSS-POL
C
          DO 221 L=1,NROW
 221        E(NROW-L+1)=RW(L)
          DO 222 L=1,NROW
 222        EPHI(NROW-L+1)=RW(L+NROW)*CEXP(CMPLX(0.,PI/2.))
            CALL SOLVE(NROW,IPS,Z,E,C)
            CALL SOLVE(NROW,IPS,Z,EPHI,CPHI)
            DO 220 L=1,NROW
```

```
            WRITE(7,*) C(L)+CPHI(L)
            ET=ET+RW(L)*C(L)+RW(L)*CPHI(L)
  220       EP=EP+RW(L+NROW)*C(L)+RW(L+NROW)*CPHI(L)
        ENDIF
        EC=UC*ET
        EX=UC*EP
        ANG(I)=THETA
        ECV(I)=CABS(EC)
        EXV(I)=CABS(EX)
        ECR=REAL(EC)
        ECI=AIMAG(EC)
        EXR=REAL(EX)
        EXI=AIMAG(EX)
        PHC(I)=ATAN2(ECI,ECR+1.E-20)/RAD
        PHX(I)=ATAN2(EXI,EXR+1.E-20)/RAD
        ECX=AMAX1(ECX,ECV(I),EXV(I))
  500 CONTINUE
      DO 600 I=1,IT
        ECV(I)=AMAX1(ECV(I),1.E-10)
        EXV(I)=AMAX1(EXV(I),1.E-10)
        ECP(I)=(ECV(I)/ECX)**2
        EDP(I)=(EXV(I)/ECX)**2
        ECP(I)=AMAX1(ECP(I),.00001)
        EDP(I)=AMAX1(EDP(I),.00001)
        ECP(I)=10.*ALOG10(ECP(I))
        EDP(I)=10.*ALOG10(EDP(I))
  600 CONTINUE
      SIGMA=(ECX**2)*4.*PI
      SIGDB=10.*ALOG10(SIGMA)
      WRITE(6,*) 'BACKSCATTER, IN DB=',SIGMA,SIGDB
      OPEN(2,FILE='RCS.DAT')
      WRITE(2,*) SIGMA,SIGDB
      CLOSE(2)
C
C PRINT FIELD POINTS
C
      DO 9000 L=1,IT
        WRITE(8,*) ANG(L),ECV(L)
 5016   FORMAT(5X,F6.1,3X,2(F8.4,3X,F7.1,3X,F7.2,3X))
 9000 CONTINUE
  900 CONTINUE
 9998 STOP
      END
      SUBROUTINE SOLVE(N,IPS,UL,B,X)
      COMPLEX UL(5000),B(250),X(250),SUM
      DIMENSION IPS(250)
      NP1=N+1
      IP=IPS(1)
      X(1)=B(IP)
      DO 2 I=2,N
```

```
      IP=IPS(I)
      IPB=IP
      IM1=I-1
      SUM=(0.,0.)
      DO 1 J=1,IM1
      SUM=SUM+UL(IP)*X(J)
    1 IP=IP+N
    2 X(I)=B(IPB)-SUM
      K2=N*(N-1)
      IP=IPS(N)+K2
      X(N)=X(N)/UL(IP)
      DO 4 IBACK=2,N
      I=NP1-IBACK
      K2=K2-N
      IPI=IPS(I)+K2
      IP1=I+1
      SUM=(0.,0.)
      IP=IPI
      DO 3 J=IP1,N
      IP=IP+N
    3 SUM=SUM+UL(IP)*X(J)
    4 X(I)=(X(I)-SUM)/UL(IPI)
      RETURN
      END
      SUBROUTINE DECOMP(N,IPS,UL)
      COMPLEX UL(5000),PIVOT,EM
      DIMENSION SCL(250),IPS(250)
      DO 5 I=1,N
      IPS(I)=I
      RN=0.
      J1=I
      DO 2 J=1,N
      ULM=ABS(REAL(UL(J1)))+ABS(AIMAG(UL(J1)))
      J1=J1+N
      IF(RN-ULM) 1,2,2
    1 RN=ULM
    2 CONTINUE
      SCL(I)=1./RN
    5 CONTINUE
      NM1=N-1
      K2=0
      DO 17 K=1,NM1
      BIG=0.
      DO 11 I=K,N
      IP=IPS(I)
      IPK=IP+K2
      SIZE=(ABS(REAL(UL(IPK)))+ABS(AIMAG(UL(IPK))))*SCL(IP)
      IF(SIZE-BIG) 11,11,10
   10 BIG=SIZE
      IPV=I
```

```
   11 CONTINUE
      IF(IPV-K) 14,15,14
   14 J=IPS(K)
      IPS(K)=IPS(IPV)
      IPS(IPV)=J
   15 KPP=IPS(K)+K2
      PIVOT=UL(KPP)
      KP1=K+1
      DO 16 I=KP1,N
      KP=KPP
      IP=IPS(I)+K2
      EM=-UL(IP)/PIVOT
   18 UL(IP)=-EM
      DO 16 J=KP1,N
      IP=IP+N
      KP=KP+N
      UL(IP)=UL(IP)+EM*UL(KP)
   16 CONTINUE
      K2=K2+N
   17 CONTINUE
      RETURN
      END
      SUBROUTINE ZMATWW(NM,A,B,NT,XT,AT,Z)
C
C *** MODS FOR LOOP -- USING HARRINGTON'S TEXT BOOK EQUATIONS AS A
C CHECK FOR OTHER METHODS. NM IS THE NUMBER OF AZIMUTHAL MODES
USED
C
      COMPLEX Z(5000),CON,CMPLX,FK
      DIMENSION XT(300),AT(300)
      PI=3.1415926
      BK=2.*PI
      CON=CMPLX(0.,PI*377.*BK*B)
      NROW=2.*NM+1
      DO 10 I=-NM,NM
        DO 10 J=-NM,NM
        IJ=J+NM+1+(I+NM)*NROW
        Z(IJ)=(0.,0.)
10    CONTINUE
C
C ONLY DIAGONAL ELEMENTS ARE NONZERO.  ALTHOUGH SYMMETRY EXISTS
BETWEEN
C Z(-N,-N) AND Z(N,N) IT IS NOT BEING USED.
C
      DO 20 I=-NM,NM
        J=I+NM+1+(I+NM)*NROW
        IP=I+1
        IM=I-1
        Z(J)=(.5*FK(IM,B,A,NT,XT,AT)+.5*FK(IP,B,A,NT,XT,AT)-
     *  (I/BK/B)**2*FK(I,B,A,NT,XT,AT))*CON
```

```
   20  CONTINUE
       RETURN
       END
       SUBROUTINE PLANEW(N,B,THR,PHR,R)
C
C  PLANE WAVE RECEIVE VECTOR ELEMENTS FOR WIRE USING THE
C  FORMULATION FROM HARRINGTON'S BOOK.  N IS THE NUMBER OF
C  AZIMUTHAL MODES. NOTE THAT B(N)=R(-N) (B IS EXCITATION AND
C  R IS RECEIVE).
C
       COMPLEX R(1000),CEXP,EXP,CMPLX
       PI=3.14159
       BK=2.*PI
       CT=COS(THR)
       ST=SIN(THR)
       RR=BK*B*ST
       NROW=2*N+1
C DO THETA RECEIVE COMPONENTS FIRST
       DO 10 I=-N,N
         IP=I+1
         IM=I-1
         EXP=CEXP(CMPLX(0.,I*PHR))
         R(I+N+1)=-PI*B*(0.,1.)**I*EXP*(BESSJ(IP,RR)+BESSJ(IM,RR))*CT
   10  CONTINUE
C NOW DO PHI RECEIVE COMPONENTS
       DO 20 I=-N,N
         IP=I+1
         IM=I-1
         EXP=CEXP(CMPLX(0.,I*PHR))
         R(I+N+1+NROW)=PI*B*(0.,1.)**(I+1)*EXP*(BESSJ(IP,RR)
     *   -BESSJ(IM,RR))
   20  CONTINUE
       RETURN
       END
       COMPLEX FUNCTION FK(N,B,A,NT,XT,AT)
       COMPLEX SUM,EXP1,EXP2,CEXP,CMPLX
       DIMENSION XT(300),AT(300)
       PI=3.14159
       BK=2.*PI
       CC=1./BK
       P1=(2.*PI-0.)/2.
       P2=(2.*PI+0.)/2.
       SUM=(0.,0.)
       DO 10 I=1,NT
         PHI=P1*XT(I)+P2
         RR=SQRT(4.*SIN(PHI/2.)**2+(A/B)**2)
         EXP1=CEXP(CMPLX(0.,-BK*B*RR))
         EXP2=CEXP(CMPLX(0.,-N*PHI))
         SUM=SUM+AT(I)*EXP1*EXP2/RR
   10  CONTINUE
```

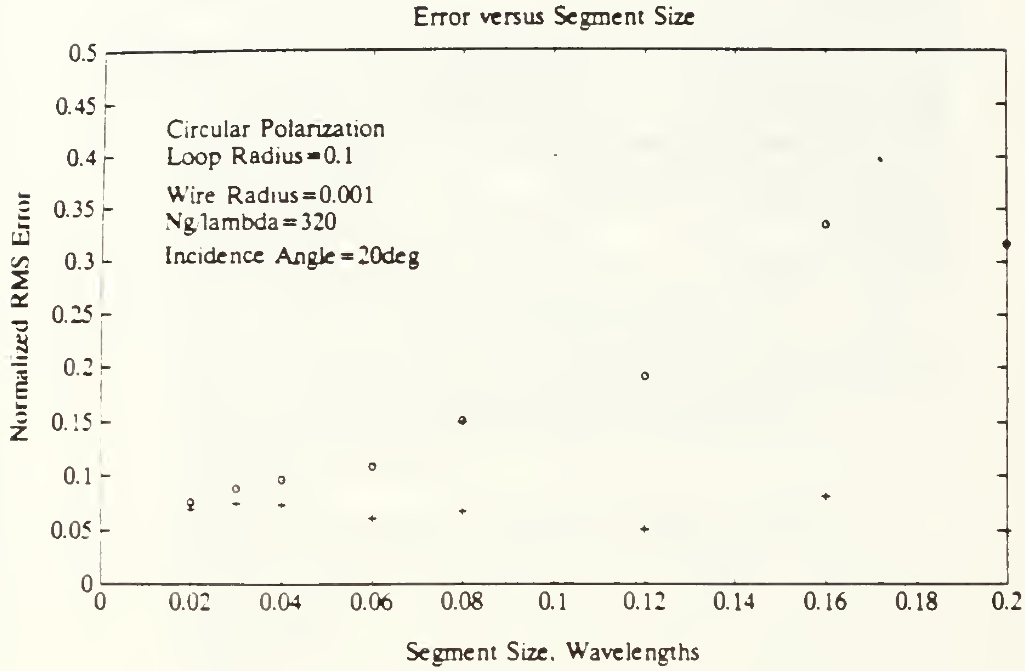
```
      FK=SUM*P1*CC
      RETURN
      END
      FUNCTION BESSJ(NN,X)
C RETURNS THE BESSEL FUNCTION B OF ORDER N (>1) AND REAL
C ARGUMENT X.
      PARAMETER (IACC=40,BIGNO=1.E10,BIGNI=1.E-10)
      IF(NN.LT.0) N=-NN
      IF(NN.GE.0) N=NN
      KC=3
      IF(N.EQ.0) KC=1
      IF(N.EQ.1) KC=2
      GO TO (1,2,3),KC
1     BESSJ=BESSJ0(X)
      GO TO 4
2     BESSJ=BESSJ1(X)
      GO TO 4
3     BESSJ=0.
      IF(ABS(X).LT.1.E-5) GO TO 4
      TOX=2./X
      IF(X.GT.FLOAT(N)) THEN
        BJM=BESSJ0(X)
        BJ=BESSJ1(X)
        DO 11 J=1,N-1
          BJP=J*TOX*BJ-BJM
          BJM=BJ
          BJ=BJP
   11   CONTINUE
        BESSJ=BJ
      ELSE
        M=2*((N+INT(SQRT(FLOAT(IACC*N))))/2)
        BESSJ=0.
        JSUM=0.
        SUM=0.
        BJP=0.
        BJ=1.
        DO 12 J=M,1,-1
          BJM=J*TOX*BJ-BJP
          BJP=BJ
          BJ=BJM
          IF(ABS(BJ).GT.BIGNO) THEN
            BJ=BJ*BIGNI
            BJP=BJP*BIGNI
            BESSJ=BESSJ*BIGNI
            SUM=SUM*BIGNI
          ENDIF
          IF(JSUM.NE.0) SUM=SUM+BJ
          JSUM=1-JSUM
          IF(J.EQ.N) BESSJ=BJP
   12   CONTINUE
```

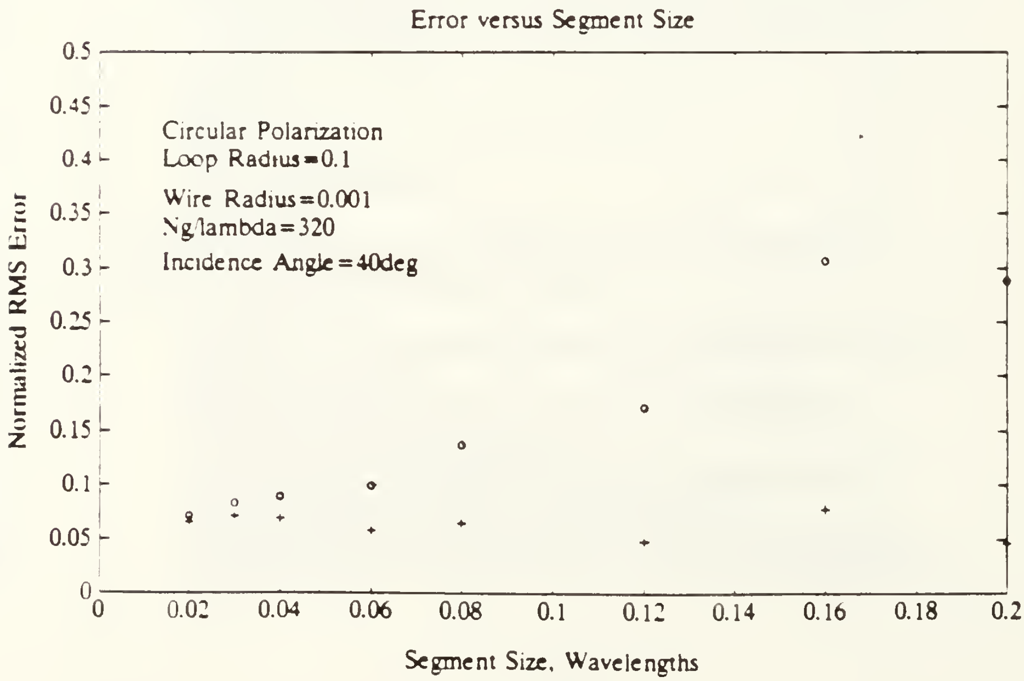
```
        SUM=2.*SUM-BJ
        BESSJ=BESSJ/SUM
      ENDIF
4     CONTINUE
      IF(NN.LT.0) BESSJ=(-1.)**N*BESSJ
      RETURN
      END
      FUNCTION BESSJ0(X)
C
C BESSEL FUNCTION OF 0 ORDER, REAL ARGUMENT X
C (SEE 'NUMERICAL RECIPES', P.172)
C
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     * S1,S2,S3,S4,S5,S6
      DATA P1,P2,P3,P4,P5/1.D0,-.109862827D-2,.2734510407D-4,
     * -.2073370639D-5,.2093887211D-6/
      DATA Q1,Q2,Q3,Q4,Q5/-.1562499995D-1,.1430488765D-3,
     * -.6911147651D-5,.7621095161D-6,-.934945152D-7/
      DATA R1,R2,R3,R4,R5,R6/57568490574.D0,-13362590354.D0,
     * 651619640.7D0,-11214424.18D0,77392.33017D0,-184.9052456D0/
      DATA S1,S2,S3,S4,S5,S6/57568490411.D0,1029532985.D0,
     * 9494680.718D0,59272.64853D0,267.8532712D0,1.D0/
      IF(ABS(X).LT.8.) THEN
      Y=X**2
      BESSJ0=(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))/
     *    (S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
      AX=ABS(X)
      Z=8./AX
      Y=Z**2
      XX=AX-.785398164
      BESSJ0=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+
     *    Y*(P4+Y*P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+
     *    Y*(Q4+Y*Q5)))))
      ENDIF
      RETURN
      END
      FUNCTION BESSJ1(X)
C
C BESSEL FUNCTION B OF ORDER 1, REAL ARGUMENT X
C (SEE 'NUMERICAL RECIPES', P.173)
C
      REAL*8 Y,P1,P2,P3,P4,P5,Q1,Q2,Q3,Q4,Q5,R1,R2,R3,R4,R5,R6,
     * S1,S2,S3,S4,S5,S6
      DATA P1,P2,P3,P4,P5/1.D0,.183105D-2,-.3516396496D-4,
     * .2457520174D-5,-.20337019D-6/
      DATA Q1,Q2,Q3,Q4,Q5/.04687499995D0,-.2002690873D-3,
     * .8449199096D-5,-.99228987D-6,.105787412D-6/
      DATA R1,R2,R3,R4,R5,R6/72362614232.D0,-7895059235.D0,
     * 242396853.1D0,-2972611.439D0,15704.4826D0,-30.16036606D0/
```

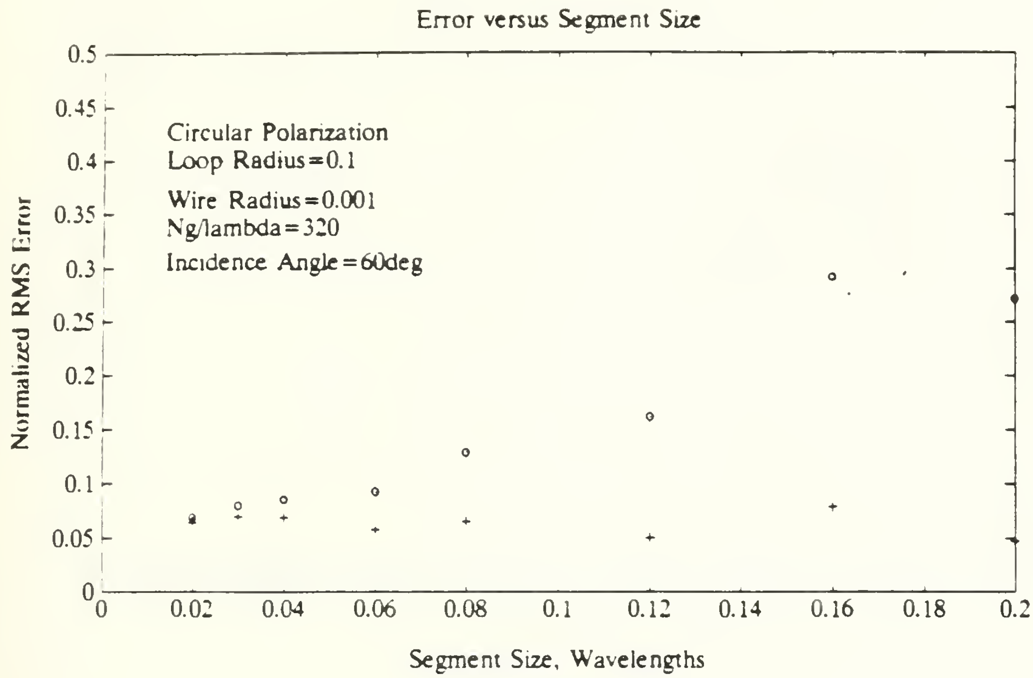
```
      DATA S1,S2,S3,S4,S5,S6/144725228442.D0,2300535178.D0,
     * 18583304.74D0,99447.43394D0,376.9991397D0,1.D0/
      IF(ABS(X).LT.8.) THEN
      Y=X**2
      BESSJ1=X*(R1+Y*(R2+Y*(R3+Y*(R4+Y*(R5+Y*R6)))))/
     *    (S1+Y*(S2+Y*(S3+Y*(S4+Y*(S5+Y*S6)))))
      ELSE
      AX=ABS(X)
      Z=8./AX
      Y=Z**2
      XX=AX-2.356194491
      BESSJ1=SQRT(.636619772/AX)*(COS(XX)*(P1+Y*(P2+Y*(P3+
     *    Y*(P4+Y*P5))))-Z*SIN(XX)*(Q1+Y*(Q2+Y*(Q3+
     *    Y*(Q4+Y*Q5)))))*SIGN(1.,X)
      ENDIF
      RETURN
      END
```

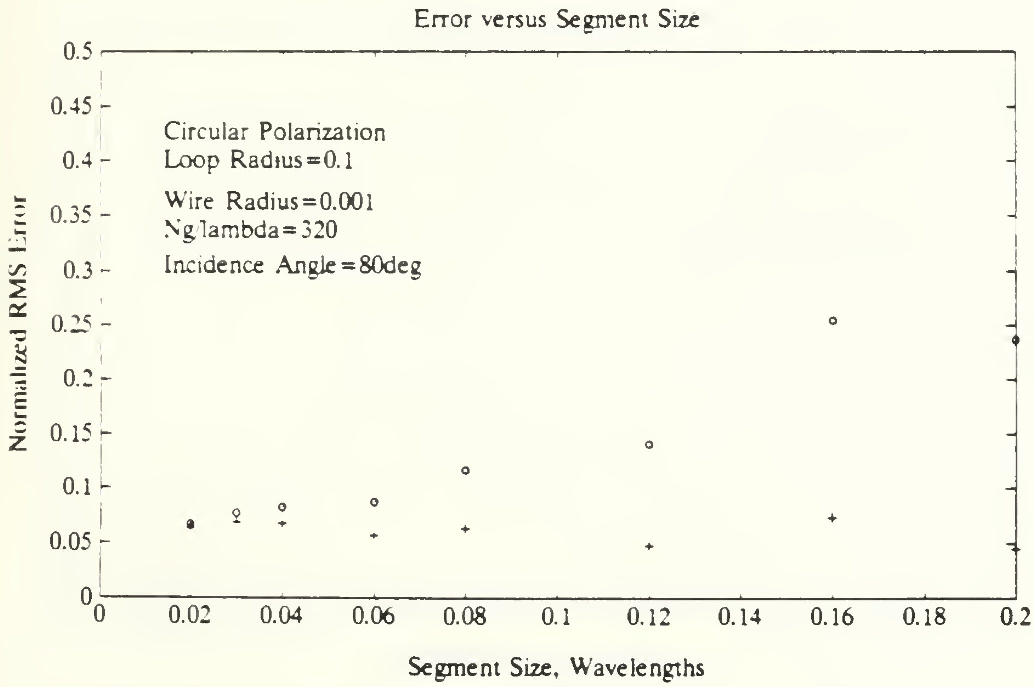
APPENDIX B

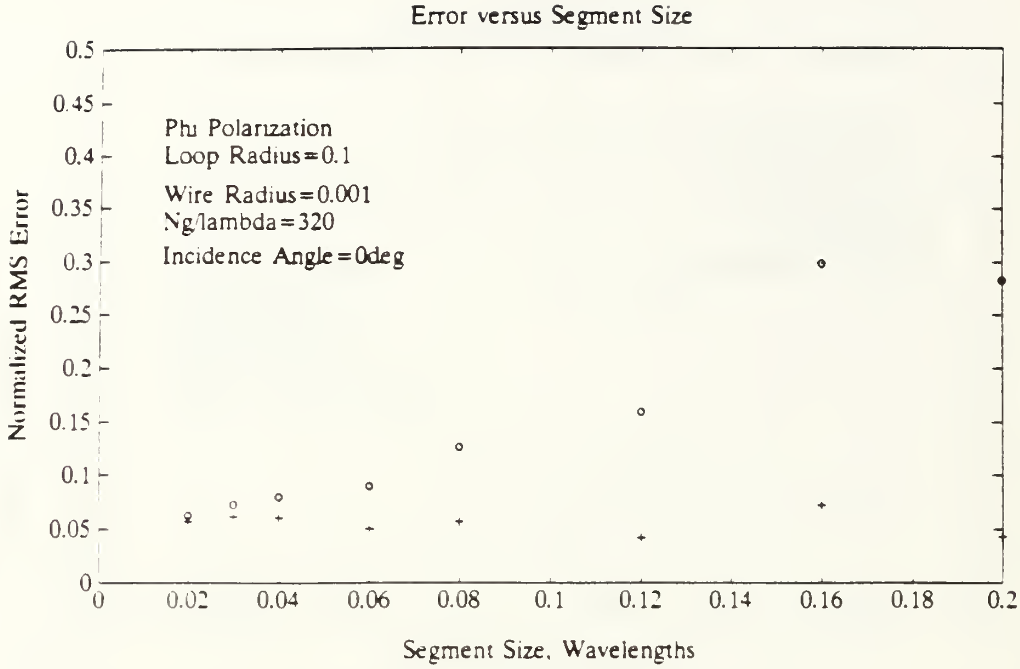
ADDITIONAL PLOTS OF CURRENT AND RMS ERROR



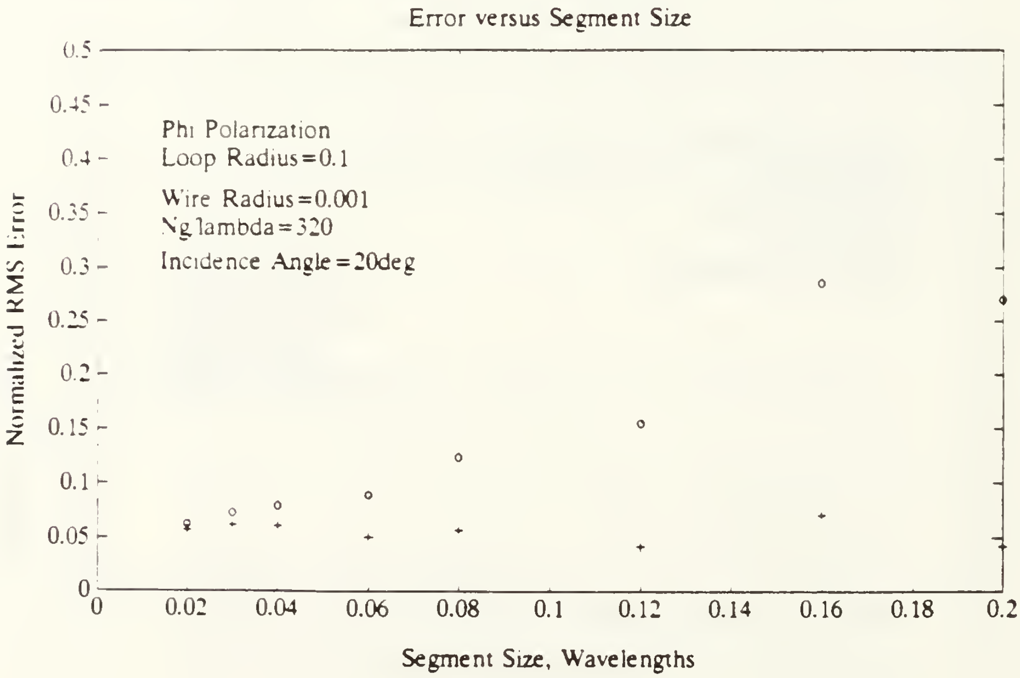



Error versus Segment Size
Circular Polarization
Loop Radius=0.1
Wire Radius=0.001
Ng/lambda=320
Incidence Angle=60deg
Normalized RMS Error
Segment Size, Wavelengths
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2

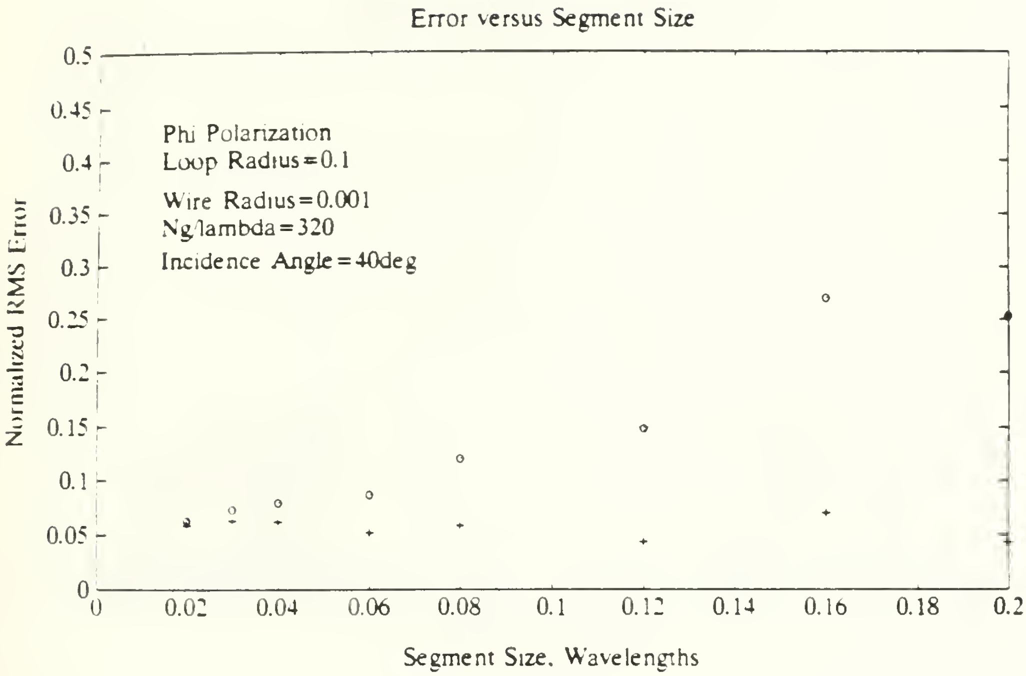

Error versus Segment Size
Circular Polarization
Loop Radius=0.1
Wire Radius=0.001
Ng/lambda=320
Incidence Angle=80deg
Normalized RMS Error
Segment Size, Wavelengths
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2

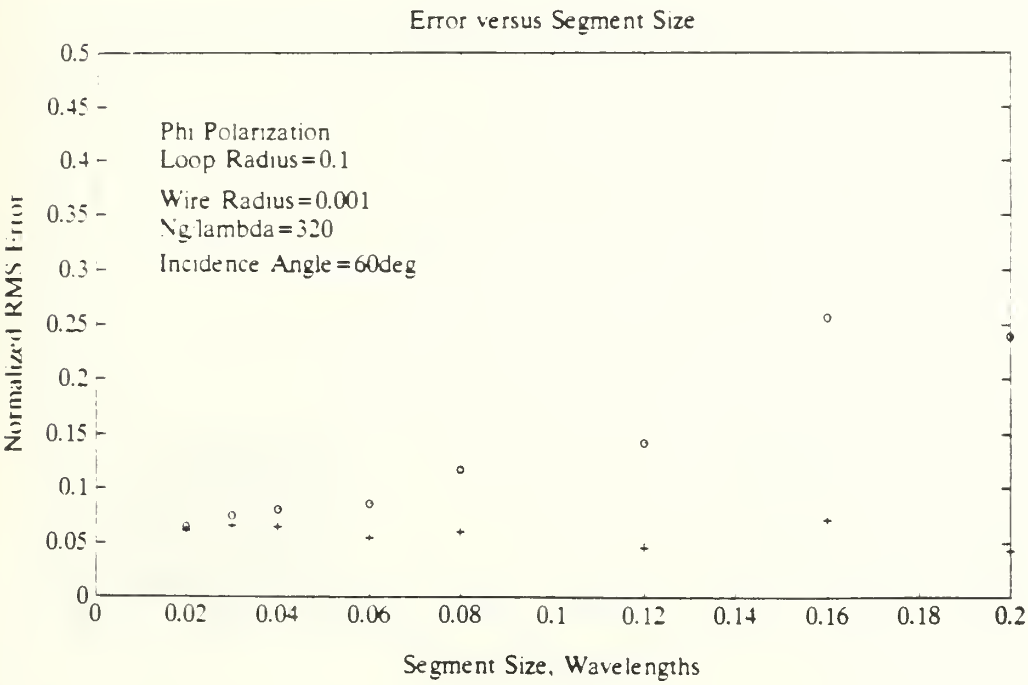


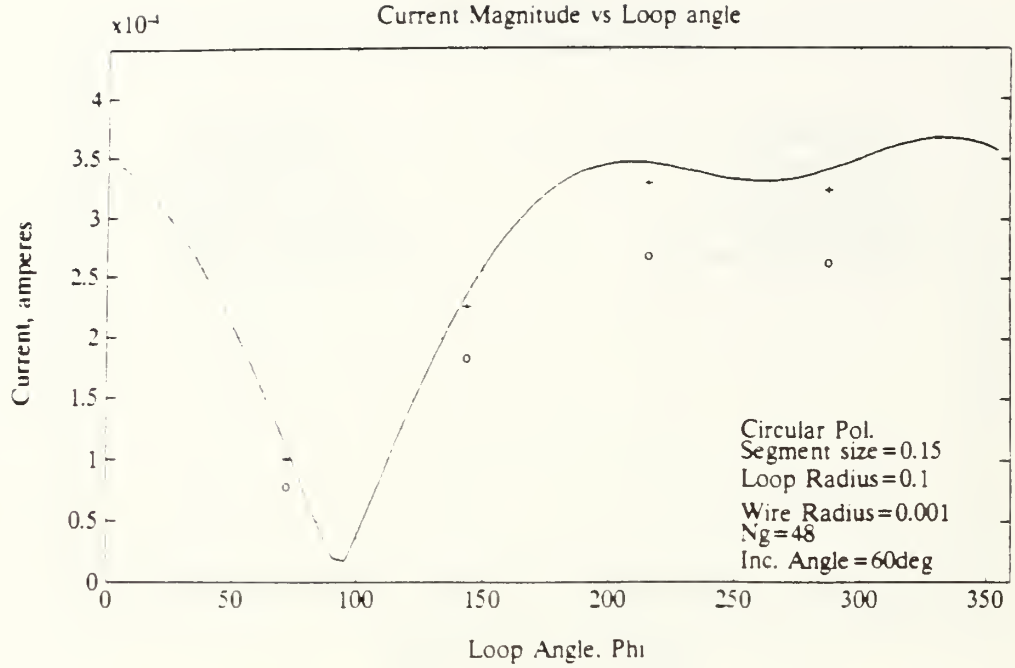
Error versus Segment Size
Phi Polarization
Loop Radius=0.1
Wire Radius=0.001
Ng/lambda=320
Incidence Angle=0deg
Normalized RMS Error
Segment Size, Wavelengths
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2



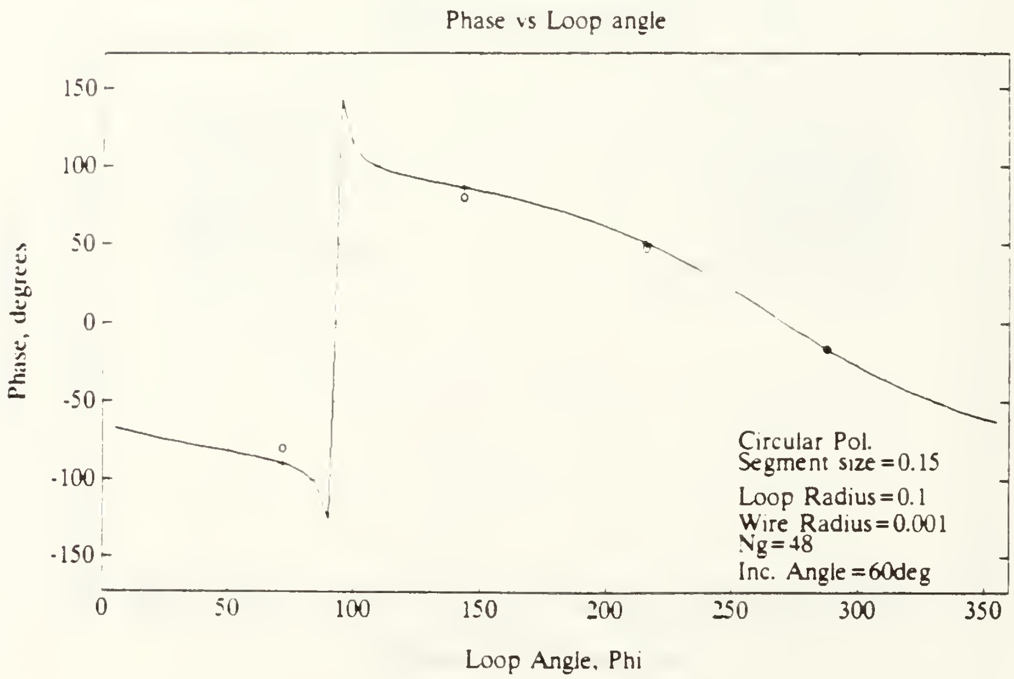
Error versus Segment Size
Phi Polarization
Loop Radius=0.1
Wire Radius=0.001
Ng/lambda=320
Incidence Angle=20deg
Normalized RMS Error
Segment Size, Wavelengths
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2


Error versus Segment Size
Phi Polarization
Loop Radius=0.1
Wire Radius=0.001
Ng/lambda=320
Incidence Angle=40deg
Normalized RMS Error
Segment Size, Wavelengths
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2

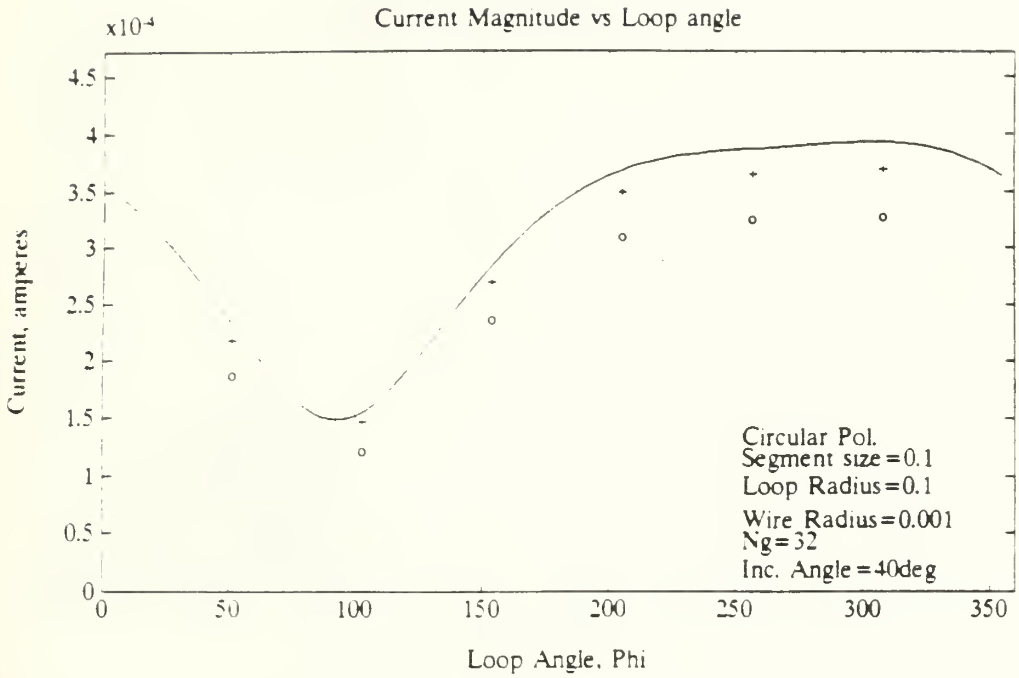

Error versus Segment Size
Phi Polarization
Loop Radius=0.1
Wire Radius=0.001
Ng/lambda=320
Incidence Angle=60deg
Normalized RMS Error
Segment Size, Wavelengths
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2



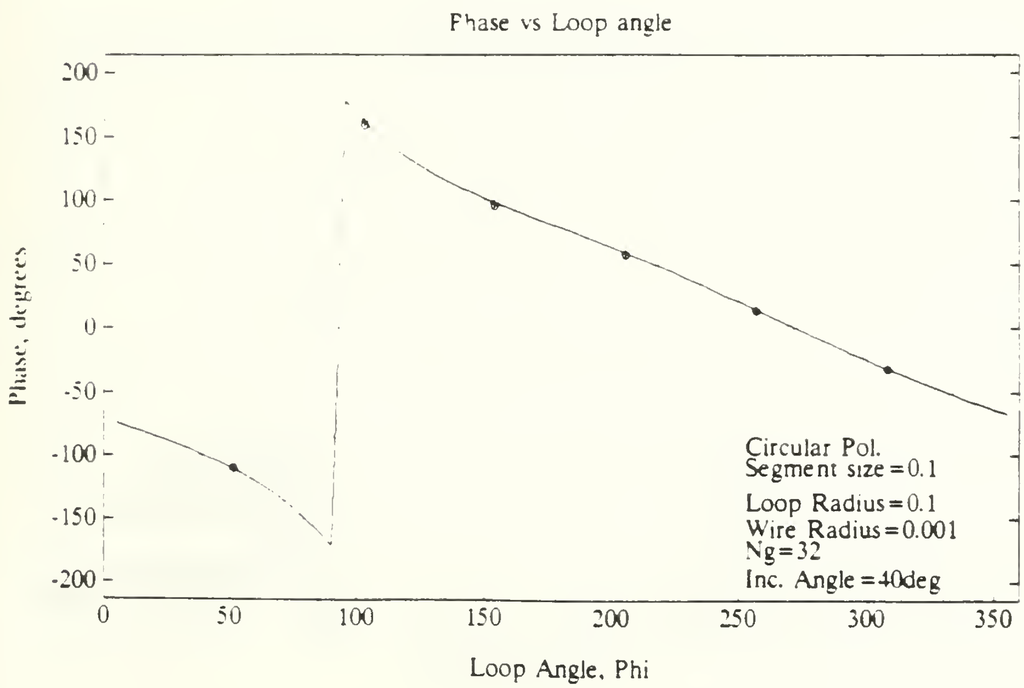
Current Magnitude vs Loop angle
x10⁻⁴
Current, amperes
Loop Angle, Phi
Circular Pol.
Segment size=0.15
Loop Radius=0.1
Wire Radius=0.001
Ng=48
Inc. Angle=60deg



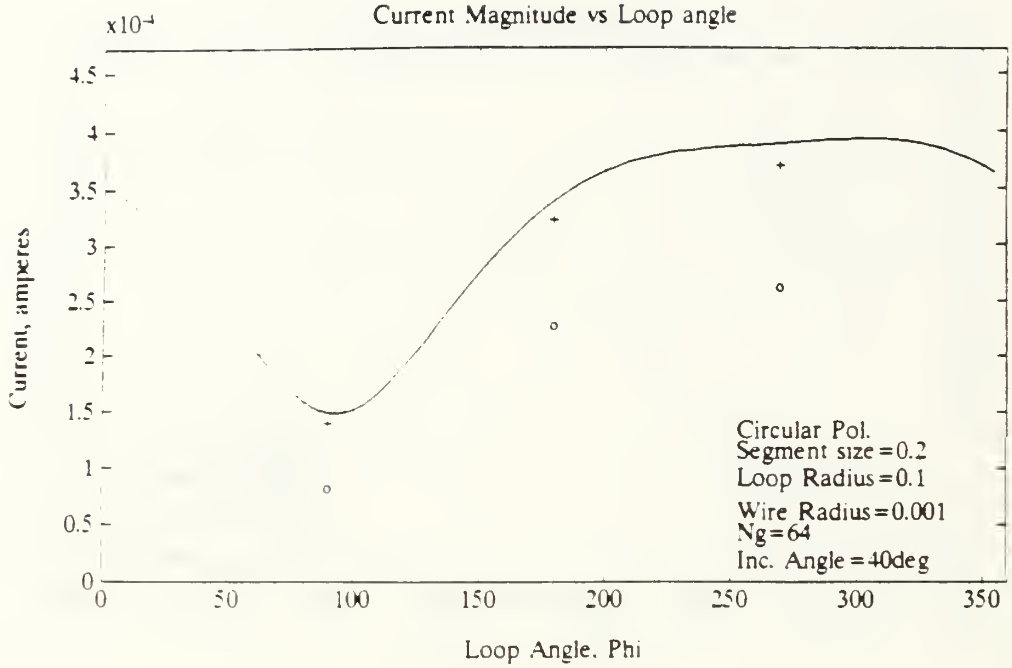
Phase vs Loop angle
Phase, degrees
Loop Angle, Phi
Circular Pol.
Segment size=0.15
Loop Radius=0.1
Wire Radius=0.001
Ng=48
Inc. Angle=60deg

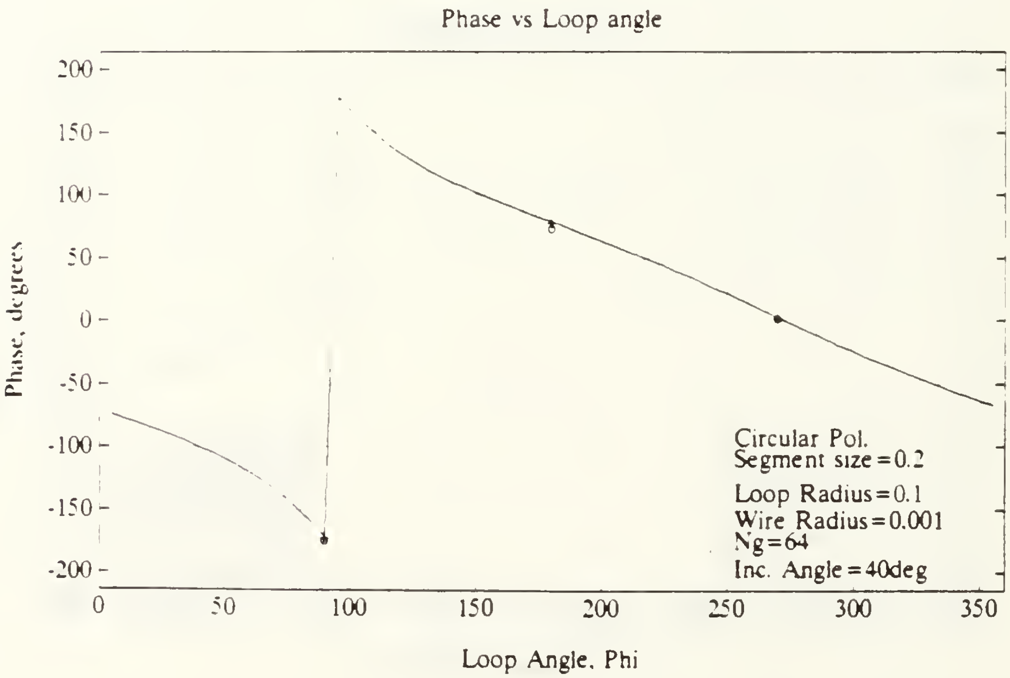


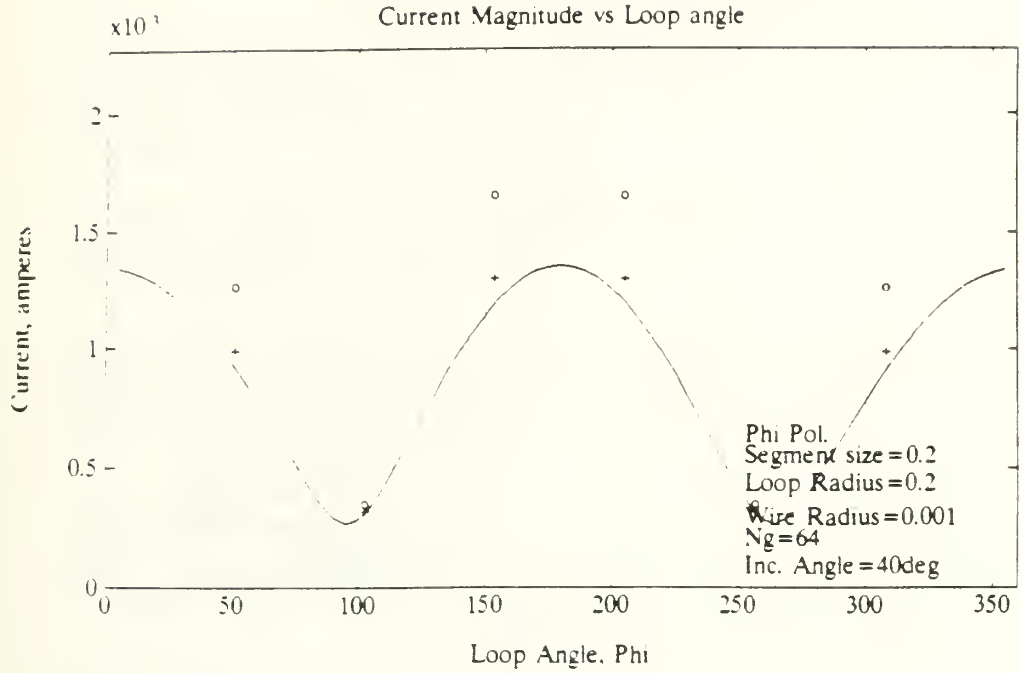
Current Magnitude vs Loop angle
x10-4
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Current, amperes
0
50
100
150
200
250
300
350
Loop Angle, Phi
Circular Pol.
Segment size=0.1
Loop Radius=0.1
Wire Radius=0.001
Ng=32
Inc. Angle=40deg



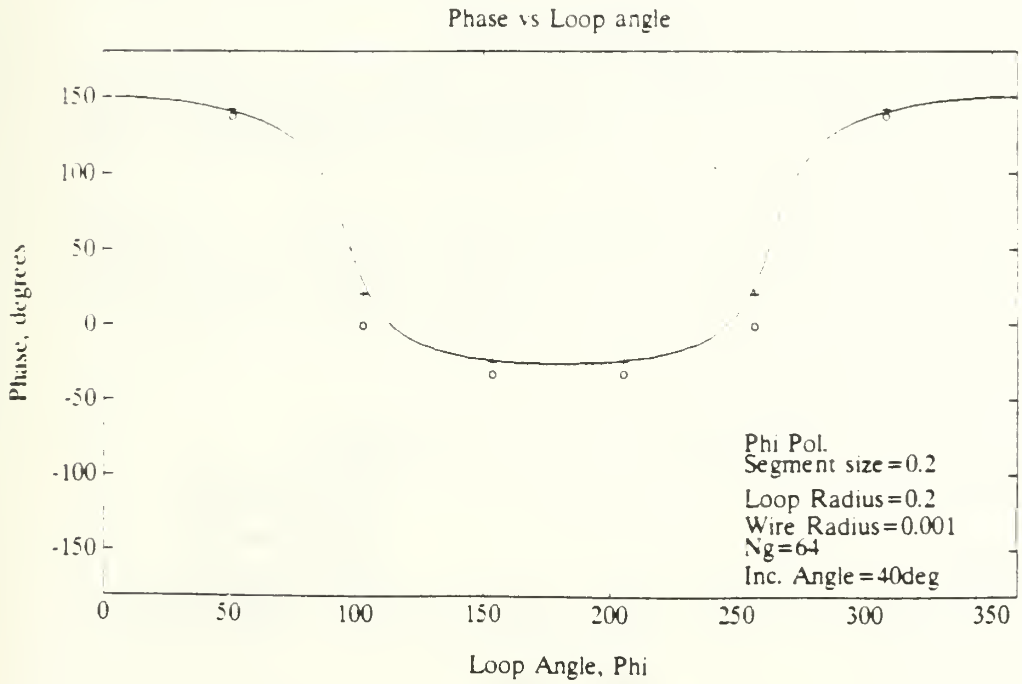
Phase vs Loop angle
200
150
100
50
0
-50
-100
-150
-200
Phase, degrees
0
50
100
150
200
250
300
350
Loop Angle, Phi
Circular Pol.
Segment size=0.1
Loop Radius=0.1
Wire Radius=0.001
Ng=32
Inc. Angle=40deg


Current Magnitude vs Loop angle
x10^-4
Current, amperes
Loop Angle, Phi
Circular Pol.
Segment size=0.2
Loop Radius=0.1
Wire Radius=0.001
Ng=64
Inc. Angle=40deg

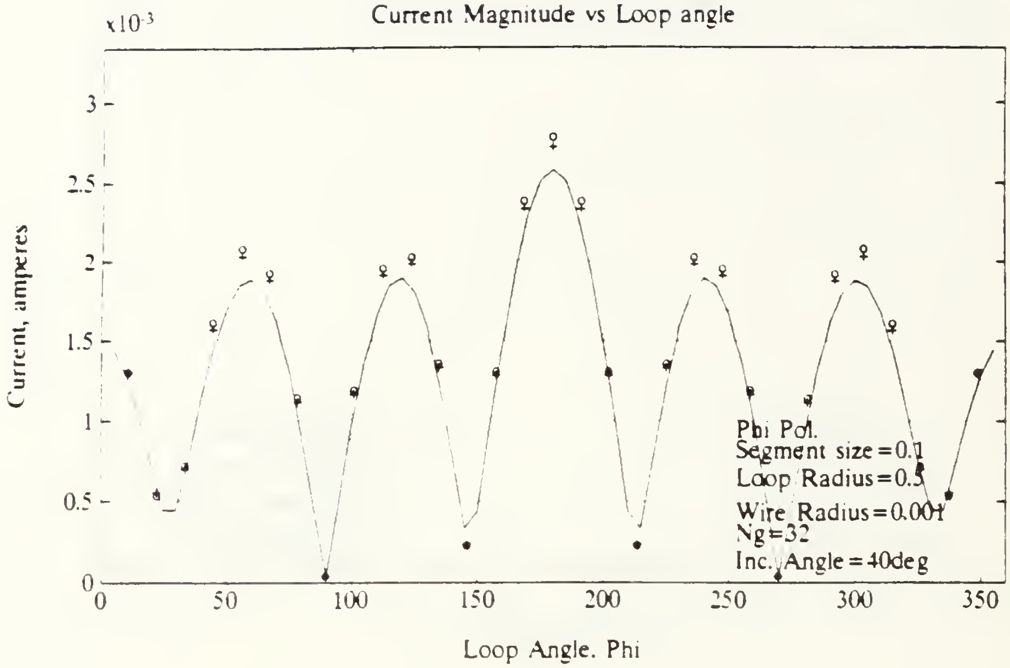

Phase vs Loop angle
Phase, degrees
Loop Angle, Phi
Circular Pol.
Segment size=0.2
Loop Radius=0.1
Wire Radius=0.001
Ng=64
Inc. Angle=40deg

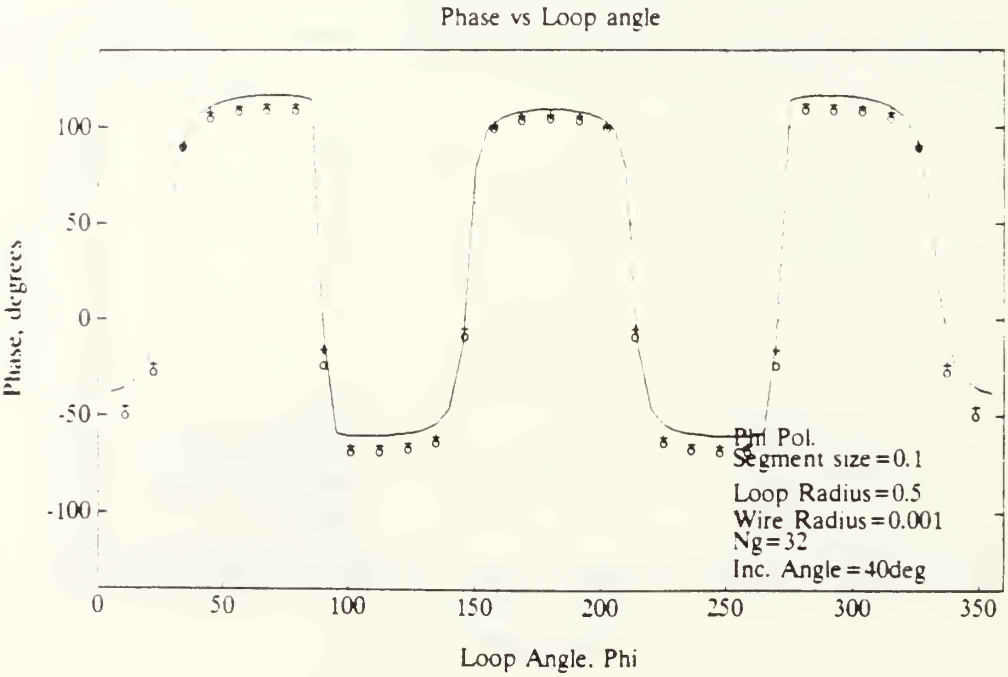


Current Magnitude vs Loop angle
x10
Current, amperes
Loop Angle, Phi
Phi Pol.
Segment size=0.2
Loop Radius=0.2
Wire Radius=0.001
Ng=64
Inc. Angle=40deg



Phase vs Loop angle
Phase, degrees
Loop Angle, Phi
Phi Pol.
Segment size=0.2
Loop Radius=0.2
Wire Radius=0.001
Ng=64
Inc. Angle=40deg


Current Magnitude vs Loop angle
x10-3
3
2.5
2
1.5
1
0.5
0
Current, amperes
0
50
100
150
200
250
300
350
Loop Angle. Phi
Phi Pol.
Segment size=0.1
Loop Radius=0.5
Wire Radius=0.001
Ng=32
Inc. Angle=40deg


Phase vs Loop angle
100
50
0
-50
-100
Phase, degrees
0
50
100
150
200
250
300
350
Loop Angle. Phi
Phi Pol.
Segment size=0.1
Loop Radius=0.5
Wire Radius=0.001
Ng=32
Inc. Angle=40deg

INITIAL DISTRIBUTION LIST

1.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
2.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
3.	Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5002	1
4.	Commander Naval Air Test Center SY90 Mr. L. Krouse Patuxent River, MD 20670-5304	1
5.	Commander Naval Air Test Center SY95BW Mr. B. Walter Patuxent River, MD 20670-5304	1
6.	Professor D. C. Jenn, Code EC/Jn Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5002	1
7.	Michael A. Morgan, Code EC/Mw Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5002	1

Thesis
W22415 Walter
c.1 The use of conformal
subdomain basis functions
in the method of moments
computations for a thin
wire.